CONTROLLED RADICAL POLYMERIZATIONS IN MINIEMULSIONS: ADVANCES IN THE USE OF RAFT

A Dissertation
Presented to
The Academic Faculty

By

James P. Russum

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in
Chemical Engineering

Georgia Institute of Technology

December 2005

CONTROLLED RADICAL POLYMERIZATIONS IN MINIEMULSIONS: ADVANCES IN THE USE OF RAFT

Approved by:

Dr. Christopher W. Jones
School of Chemical & Biomolecular Engineering
Georgia Institute of Technology

Dr. Pradeep K. Agrawal
School of Chemical & Biomolecular Engineering
Georgia Institute of Technology

Dr. F. Joseph Schork
School of Chemical & Biomolecular Engineering
Georgia Institute of Technology

Dr. J. Carson Meredith
School of Chemical & Biomolecular Engineering
Georgia Institute of Technology

Dr. Marcus Weck
School of Chemistry & Biochemistry
Georgia Institute of Technology

Date Approved: October 20, 2005

We have all heard of discoveries made by chance, and many of them, I imagine, were made by men who had already approached near to them by design.

-G. K. Chesterton

ACKNOWLEDGEMENTS

My sincere thanks go out to Chris Jones and Joseph Schork for taking me on as a student in the first place and guiding me through this process to the end. Your support and direction are greatly appreciated, and your insights, both scientific and otherwise, have proven invaluable to me. To Marcus Weck, for all the useful discussions during our group meetings and especially for the sort of positive character development that can only come from being forced to extend oneself beyond the point at which one starts to feel uncomfortable. I am certain that it will serve me well. I am deeply grateful also to Pradeep Agrawal and Carson Meredith for agreeing to be on my committee and freely offering their time and expertise.

Thanks go out to Pete Lovell for extremely enlightening and useful conversations regarding the use of NMR to characterize polymers. Alessandro Butté and Wilfred Smulders are both extended my gratitude for their able assistance, their willingness to take part in my intellectual development, and for their friendship. I owe thanks to several undergraduates, Stephen Fisher, Nick Barbre, Dennis Kalman, Eduardo Oliver, and Joel Keene, without whose time and energy (and at moments, comic relief) this would simply have been impossible. I think also that it would have been extremely difficult had I not had the support, help and friendship of the original members of the Jones group, Mike McKittrick, a.k.a. “Big Mike”, Joe Nguyen , a.k.a. “Smokin’ Joe”, and Kunquan Yu, a.k.a. “Quin”. They managed to make my tenure here both enlightening and entertaining. Many thanks go out to Michael Murphy, Wei Sen Wong and Todd Grey of Viscotek for all of the technical assistance with the GPC that they rendered.

Most importantly, if for no other reason than for having to put up with me for the last couple of years, I think God has a special place reserved in His heaven for my wife, Maria. Her patience, humor and humility kept my feet firmly planted on terra firma and helped me to remain focused on those things that are truly valuable in this life. Thank you, sweetheart, I am forever in your debt.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

$\mathbf{D}$	Diffusion coefficient, $cm^2 \cdot s$
D_j	Droplet j
d_i	Inner diameter of tube, cm; $d_{i,0}$, initial inner diameter of tubing
$\mathbf{E}_\theta$	Exit age distribution of the fluid in terms of the dimensionless residence time
f	Initiator efficiency
$\mathbf{F}$	Exit age distribution of the fluid, $\int \mathbf{E}_\theta \, dt$
$G(V_e)$	GPC chromatogram, V_e is retention volume
I	Initiator concentration, mol/L; I_j^D of droplet j, I_i^P , of plug i
k_{add}, $k_{-\beta}$	RAFT addtion rate constants, $L \cdot mol^{-1} \cdot s^{-1}$
k_{-add}, k_β	RAFT fragmentation rate constants, s^{-1}
k_d	Initiator decomposition rate constant, s^{-1}
k_p	Propagation rate constant, $L \cdot mol^{-1} \cdot s^{-1}$
k_t	Termination rate constant, $L \cdot mol^{-1} \cdot s^{-1}$
k_{trm}	Chain transfer to monomer rate constant, $L \cdot mol^{-1} \cdot s^{-1}$
L	Length, cm; L_r, reactor length, L_p, plug length, L_g, gap length
M	Molecular weight of polymer
M_n	Number average molecular weight
M_w	Weight average molecular weight
n_d	Number of droplets
n_p	Number of plugs
$N(M)$	Cumulative number molecular weight distribution
P_i	Plug i
q_D	Volumetric flow rate of droplets, $cm^3 \cdot s^{-1}$
Re	Reynolds number, dimensionless, $d_i u \rho / \mu$
R_{RMS}	Root mean square radius
t	Time, s; t_d, time to traverse one droplet length, t_g, time to traverse one gap length, t_p, time to traverse one plug length
$\bar{t}$	Average residence time of fluid in reactor, s; V/v
T_g	Glass transition temperature, K
u	Average bulk fluid velocity, $cm \cdot s^{-1}$
v	Volumetric flow rate of fluid in flow reactor, $cm^3 \cdot s^{-1}$
V	Volume, cm^3; V_0, initial plug volume, V_D, droplet volume, V_i^P , volume plug i, $V_{P,1}$, volume of plug 1 cm in length at d_{i0}
X	Conversion; X_i, conversion plug i

Greek letters

μ	Absolute viscosity, P
ρ	Density, $gm \cdot cm^{-3}$

σ_θ	Variance of $\mathbf{E}_\theta$ curve, dimensionless
τ	Average residence time in reactor, s
θ	Dimensionless residence time, t/τ

SUMMARY

The goal of this work is to increase the current understanding of Controlled Radical Polymerizations (CRPs) in two areas. Progressing closer towards employing an aqueous system, specifically miniemulsion, to produce poly(vinyl acetate) via RAFT chemistry constitutes the first part of this goal. Presented are the results of miniemulsion polymerizations using both water and oil-soluble initiators. Limiting conversions in both are examined and explained in terms of radical loss. The second part of the goal is to further the understanding of the nature of the RAFT/miniemulsion system when employed in continuous tubular reactors. The development of the recipe using mixed surfactants, the results of styrene homopolymerizations in batch and tube, and the results of a chain extension experiment demonstrating the living nature of the chains formed in the tubular reactor are presented. Kinetic anomalies are addressed, as well as polydispersity differences between batch and tube. Flow phenomenon and their influence on residence time distribution and by implication the PDI of the polymer formed are offered as explanations for the variance in PDI and are subsequently quantified. A model of RAFT in laminar flow is presented and the results and implications are discussed in general terms. The flow profile of the reactor is examined using a tracer technique developed specifically for this system. Experiments are presented directly relating the residence time distribution to the polydispersity of the polymer. Transient behavior of the reactor in isolated plug flow is explained in terms of initiator loss. Both experimental data and a model are used to support this hypothesis. Finally, conclusions and implications are

presented and unanswered questions and the ideas for future work that they generated are addressed.

CHAPTER 1

INTRODUCTION

1.1 Controlled Radical Polymerizations (CRP)

1.1.1 Background

The advent of controlled radical polymerization (CRP)[1] has provided a potential means by which polymers of well defined architecture can be synthesized using more forgiving and robust free-radical chemistry. Before CRPs were developed, little could be done with free radical processes to tune the molecular weight or the fine structure of homo and copolymers. Because it is a free radical process, it has relatively mild reaction conditions and is tolerant to the presence of small amounts of impurities such as air, water and trace chemicals. Since by some estimates upwards of 50% of polymers are produced industrially via some free radical process,[2] CRPs offer a possible solution to synthesizing distinctive homo and copolymers on a large scale economically. General features of CRPs include the following:

1. Pseudo-first order kinetics
2. Linear evolution of the number average molecular weight with conversion
3. Narrow molecular weight polydispersities, M_w/M_n
4. Ability of the chains to grow after the initial monomer charge is exhausted

The most important of these features by far and what makes these systems attractive is their ability to continue adding monomer units. It is in this way that structural design of the polymer can be carried out. Because dormant chains always have the potential to add

additional monomer units, CRPs allow for the relatively facile synthesis of block and other copolymers, e.g. by simply adding a second, different monomer after the charge of the first monomer has been exhausted. This presents the possibility of synthesizing a broad range of architectures with different chemical and material properties, as shown in Figure 1.1.[3]

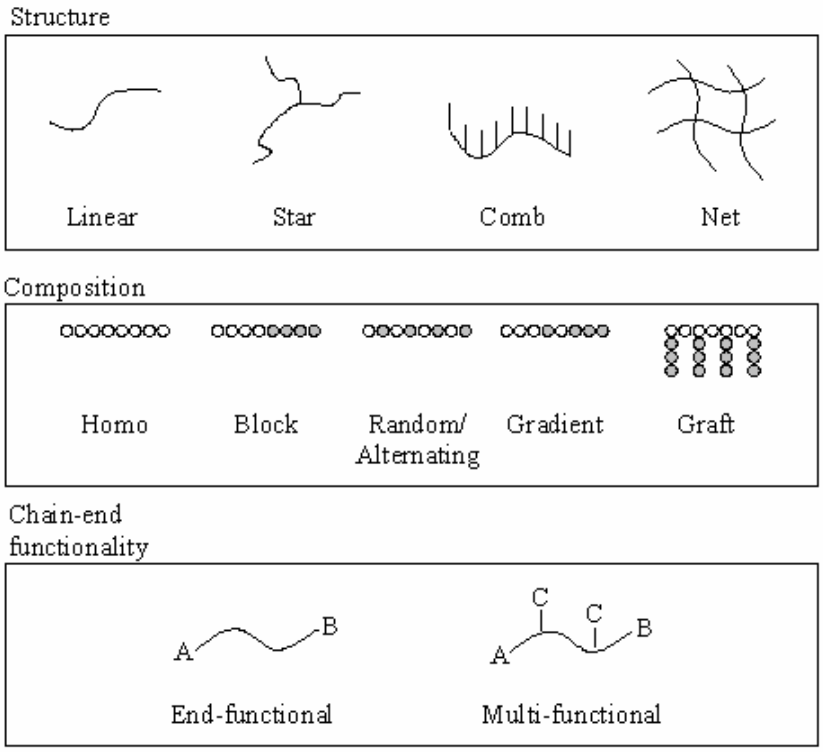


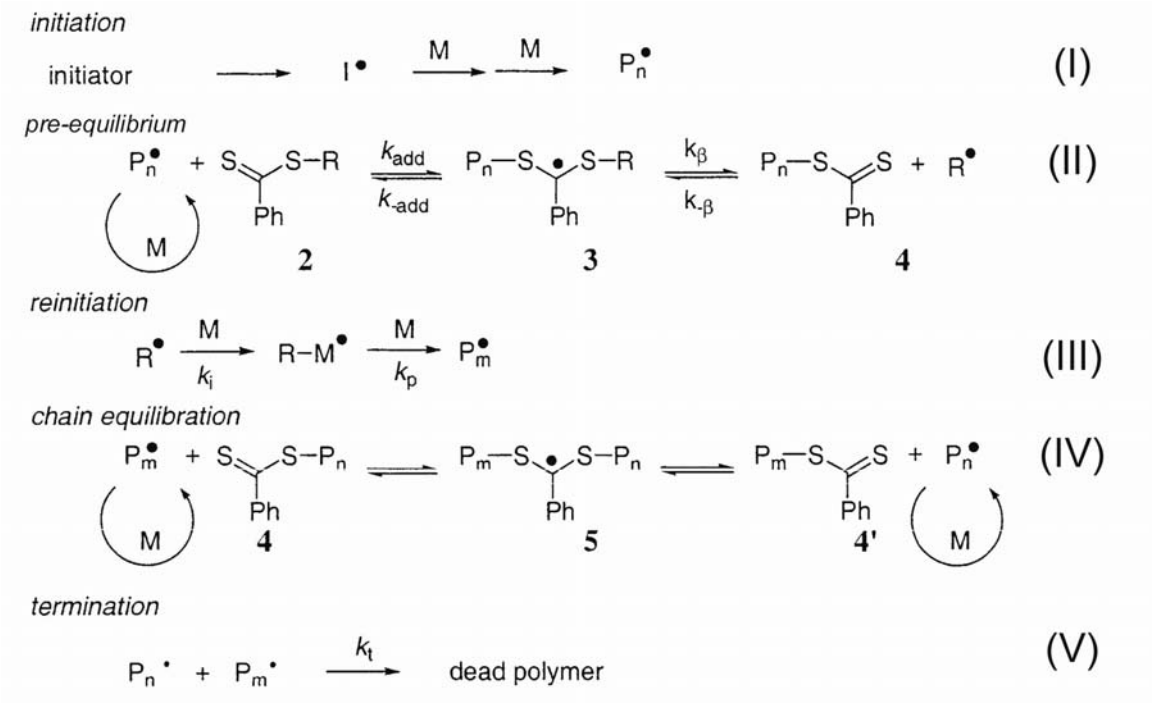
Figure 1.1. Potential architecture and functionality control afforded by CRP (adapted from [3])

While exact mechanisms vary, CRPs maintain control of the reaction by converting the propagating radicals into a temporarily dormant species thereby limiting the lifetime of the radical. As such, the number of bimolecular terminations is greatly reduced. Most CRPs can be classified under two broad categories, either reversible termination or reversible transfer. Atom transfer radical polymerization (ATRP)[4-6] and nitroxide

mediated radical polymerization (NMP)[7,8] are the two most prominent examples of reversible termination. Both techniques use a radical deactivator, a nitroxide (e.g. 2,2,6,6-tetramethylpiperidinine-N-oxyl, TEMPO) in the case of NMP, and a metal halide complex (e.g. $RuCl_2(PPh_3)_3$) in the case of ATRP, to reversibly terminate growing chain ends. The rate of the reversible termination is such that the instantaneous lifetime of a propagating radical is exceedingly small and the overall concentration of radicals is very low. Each growing chain adds only a few monomer units during its active conformation and if initiation is fast and quantitative, the molecular weight increases linearly with conversion. Since termination is second order with respect to radical concentration and the radical concentration lower than conventional free radical, the number of dead chains is greatly reduced as compared to conventional free radical reactions.

Reversible transfer utilizes a rapid exchange reaction between an active and dormant chain, such that one of the chains will always be active. In theory the transfer mechanism does not effect the radical concentration, rather the number of radicals is determined by the addition of a conventional free-radical initiator. Since each radical formed from the initiator will eventually terminate to form a dead chain, the best overall results are obtained when the concentration of initiator is small with respect to the control agent. Reversible addition-fragmentation chain transfer (RAFT, see Scheme 1.1)[9] is the most commonly utilized technique for producing living polymers via reversible transfer.[10,11] RAFT agents are typically dithio compounds because they have been found to be extremely versatile and work with a wide range of monomers. It should be noted that pseudo-first order kinetics need not necessarily apply to RAFT systems for them to be controlled.[12,13] Unlike reversible termination, in which either a nitroxide (NMP) or

halide initiator (ATRP) is utilized, traditional thermally decomposing free radical initiators are employed as the radical source in RAFT polymerizations. The role of the initiator in reversible termination is primarily to determine the number of growing



Scheme 1.1 Reactions describing Reversible Addition Fragmentation Chain Transfer (RAFT) process as commonly accepted in the literature (adapted from [14])

polymer chains (quantitative initiation),[15] whereas in RAFT polymerizations the number of growing chains is determined by the amount of RAFT agent consumed. The total number of dormant chains will be equal to the total number of RAFT agent molecules, which is equal to the number of leaving group radicals R•. If termination is by disproportionation then the total number of dead chains eventually equals the number of initiating radicals, I•. If termination is by combination, then the total number of dead

chains will be one-half the number of initiating radicals, or ½ I•. Given this condition, the ratio of initiator to RAFT should be kept as low as possible, insuring a minimum of dead chains. In practice, the proportion of dead chains can easily be kept to levels below 10% and the molecular weight of the polymer can be estimated by:

$$\bar{M}_{n,theo} = MW_{RAFT} + \frac{[\mathrm{M}]}{[\mathrm{RAFT}]} \times MW_{mon} \times X$$

The pre-equilibrium reaction (I) is extremely important in affecting the overall control of the reaction and RAFT agents with high addition rates ($k_{add} \gg k_{\beta}$) are necessary to insure that the dormant chains are quickly formed. In particular, the transfer constant for this step, commonly defined as:

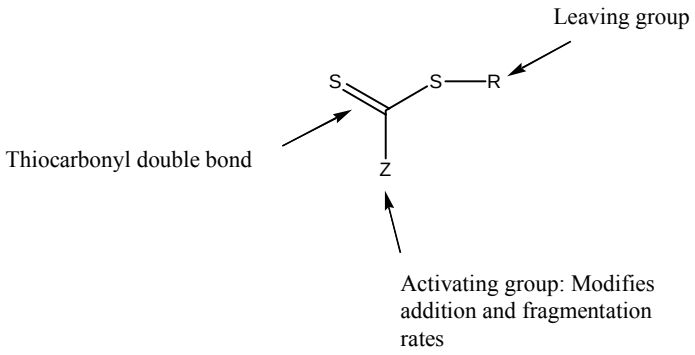
$$C_T = \frac{k_{tr}}{k_p}$$

where

$$k_{tr} = k_{add} \times \frac{k_{\beta}}{k_{\beta} + k_{-add}}$$

is critical and should be at least 10 or higher to impart a sufficient degree of control over the polymerization.[16,17] The structure of the RAFT agent, specifically the nature of the Z and R groups (see Scheme 1.2), has a profound effect upon C_T and in turn the molecular weight distribution of the resulting polymer.[14,18] The Z-group interacts with the C=S bond to enhance or impede its ability to undergo free radical addition and provides for the relative stability of the intermediate radical (Scheme 1.1, **3** and **5**). The R-group has to be

a good leaving group and capable of re-initiating the polymerization (III). Radical stability, polarity, and steric factors all contribute to the leaving ability of the R-group. Bulky, stable, electrophilic radicals make better leaving groups. If the monomer being polymerized forms a more stable radical than the R-group the reaction will not proceed. Extending this to block copolymer synthesis, the monomer that forms the most stable radical should always be polymerized first. This is because the leaving group after the



Scheme 1.2 General structure of dithiocarbonyl RAFT agent

first block is formed is the radical formed from the first monomer. If the first monomer forms a more stable radical than the second monomer (e.g., 3° vs. 2°) then the addition rate constant, k_{β}, would not exceed the fragmentation rate constant, k_{-add}, to a large enough degree for the reaction to proceed. In general, because of radical stability, monomers should be polymerized in the following order when synthesizing block copolymers:[16]

methacrylates → styrenes → acrylates

1.1.2 CRP in dispersed aqueous media

One of the obstacles to implementing CRP commercially is its ability to work in heterogeneous media, specifically dispersed aqueous systems such as suspensions or emulsions. Aqueous systems are preferred industrially because they are generally considered a "green" way of producing polymer (by way of eliminating the need for solvent), have inherently high heat transfer capacity, and result in low viscosity polymer latex that is relatively easy to manage. The primary challenge to be overcome has here to do with the partitioning of the control agent into the aqueous phase, which can result in a loss of control over the molecular weight and polydispersity. In emulsion polymerizations, highly insoluble deactivators (typical of many RAFT agents) do not transport between droplets and particles in great enough number to maintain control. On the other hand, if a control agent transports relatively easily, it can interfere with the growth of aqueous phase radicals during particle formation, slowing down and greatly complicating the kinetics. This can be the case when NMP or ATRP is applied to dispersed aqueous systems because the small, mobile nature of the nitroxide or metal complex can make them susceptible to aqueous partition. Another difficulty with CRP when applied to dispersed systems is instability of the latex, exhibited as phase separation or high levels of coagulum. Luo et al.[19] have postulated the existence of a super swelling state brought on by the presence early in controlled polymerizations of a large population of oligomers. Using a theoretical model applied to miniemulsion polymerization, they suggest that the chemical potential of particles containing oligomers is lower than that of the monomer droplets, creating a large driving force for monomer to transport to the particles and causing the colloidal instability that is often observed.

CRPs have been reported in suspensions[20-28] and emulsions,[29-39] but much of the published work has suffered from poor colloidal stability or poor control of the molecular weight and polydispersity. Additionally, solvents were required in several of the reported suspension systems, making them less attractive from an environmental and processing standpoint. While much remains to be understood before commercial implementation becomes practical, progress is being made in attempting to overcome these limitations. In one of the most successful reports of CRP in emulsion polymerization, Prescott et al.[33] showed that RAFT could be employed in a seeded emulsion by combining an organic co-solvent to facilitate the transport of the hydrophobic RAFT agent with the familiar technique of seeding the emulsion with preformed polymer seed. They employed styrene and 2-phenylprop-2-yl phenyldithioacetate (PPPDTA) in tandem with acetone as the organic co-solvent, and demonstrated that polymer with controlled characteristic (linear growth of number average molecular weight, polydispersity ~ 1.1-1.4) could indeed be produced via RAFT in emulsion. They attributed this to the enhanced transport of the RAFT agent, localizing it in the particle phase. While significant inhibition and some reduction in the rate of polymerization were observed when the concentration of RAFT agent was increased, the researchers noted that this could be overcome without significantly affecting the controlled character of the polymerization by simply increasing the level of initiator. Little or no colloidal instability was observed, however they make no mention of what factors might have contributed to this. Tsavalas et al.[40] postulated that surface activity of the RAFT agent brought on by complexation with the ionic surfactant as a possible contributor to the instability observed in mini- and macroemulsions. The successful use of nonionic surfactants or mixed surfactant systems

in overcoming instability lends credibility to the argument that some interaction between the RAFT agent and ionic surfactant may play a role, however it remains an open question.[41-43] Additionally, the research to date suggests that systems comprised of highly reactive RAFT agents can also contribute to instability, however, this can be overcome to a certain degree with the use of nonionic surfactants.[44,45] More recently, Butté et al.[37,46] have reported the successful use of cyclodextrins to facilitate the transport of the RAFT agents across the aqueous phase in emulsion polymerizations, producing well-controlled polymers and copolymers. While work is ongoing to fully understand the mechanisms involved, the current state of affairs is that successful RAFT polymerizations in dispersed media can be performed through judicious selection of reaction components.

These recent advances notwithstanding, at present the most successful CRP/dispersed aqueous system reported in the literature is CRP/miniemulsion. NMP[47-50], ATRP[51-57] and RAFT[40,41,58-68] have all been shown to work well in miniemulsions, where complications arising from the partitioning of the control agents are less of a factor than in conventional emulsion polymerizations, thus affording much better control of the polymerization. The fact that the primary locus of nucleation is the monomer droplets helps to contribute to the fast, quantitative initiation that is desirable.[69] Coagulation and colloidal instability are not as problematic in CRP/miniemulsions, provided a suitable combination of stabilizers and CRP are used. ATRP has proven the most recalcitrant in this regard, however successful reports are beginning to appear in the literature. Employing copper/ligand ATRP complexes, Li et al.[56] have suggested that only sufficiently hydrophobic ligands can maintain the activator/deactivator concentration in

the organic phase to levels needed for control. They utilized reverse-ATRP with a conventional free radical initiator (AIBN) to overcome problems associated with oxidation of Cu^{I}/ligand via exposure to air. Additionally, they have developed what they term a "simultaneous reverse and normal initiation (SNRI) ATRP process"[70] and applied it to miniemulsion polymerization.[57] They found that by employing a dual initiator system (a typical ATRP alkyl halide initiator with a more conventional free radical initiator, AIBN) that a highly active copper catalyst system [$CuBr_2$/Me_6-TREN, $CuBr_2$ complexed with 1 equivalent of tris(2-(dimethylamino)ethyl)amine)] could be used to mediate the reverse ATRP process. In principal, this opens the possibility of using more active copper catalyst complexes for reverse ATRP at temperatures that are compatible with aqueous systems. The researchers also found that nonionic surfactants (Brij 98 and HV 25) tended to be better candidates for providing stability and postulated that interference with the ATRP mechanism from the reaction of the copper complex with the surfactant might be partially responsible for an observed lack of polymerization control in ionically stabilized systems.[71]

While nonionic surfactants would seem to provide better stability and control when used with ATRP in dispersed aqueous systems, NMP has been successfully carried out in miniemulsions with conventional ionic surfactants, e.g. sodium dodecylbenzenesulfonate, or SDBS. MacLeod et al.[47] found that by tuning the ratio of surfactant (SDBS) to costabilizer (hexadecane) as well as the ratio of nitroxide (TEMPO) to initiator (potassium persulfate, KPS) stable miniemulsions resulted yielding polymer with good controlled characteristics (PDI ~ 1.1). Cunningham et. al.[49,50] have reported similar

success with NMP in miniemulsions and both groups of researchers have demonstrated the feasibility of producing copolymers via NMP in miniemulsion.[48,72]

Each of these three techniques offers its own unique set of advantages and disadvantages. The best-behaved systems in terms of termination and polydispersity are the ATRP reactions, but the oftentimes large excess of metal complexes that are employed as the control agent are left in the polymer and have to be dealt with. The residual control agent is also an issue in polymers formed via NMP and RAFT, however the compounds are not metal-based and the amounts necessary to control the polymerization tend to be less. Because the mechanism is similar, the benefits of NMP are akin to those of ATRP. At present however, temperatures (> 100 °C) required to obtain reasonable reaction rates are an issue, as well as the small number of monomers that can be polymerized with a given nitroxide. Given this limitation, a number of current efforts are aimed at developing nitroxides that can mediate the reactions at temperatures below 100 °C.[73,74] Unlike NMP, RAFT can be employed at lower reaction temperatures and will polymerize a wide range of monomers and functionalities.[44,75] While it suffers from a greater degree of termination than ATPR or NMP, as mentioned already this can be overcome in large part by keeping the molar ratio of RAFT agent to initiator sufficiently high.

Two benefits of combining CRP with miniemulsion have already been mentioned, those of predominant droplet nucleation and less dependence on transport of monomer and control agent. In this work, a RAFT/miniemulsion system was chosen because it offers two additional advantages over other CRP techniques when combined with miniemulsions. The first advantage obtains from the radical segregation effect that is

inherent to miniemulsions. In solution, any radical could terminate with another. However, when the radicals are each segregated into isolated reaction loci, e.g. the particles of a miniemulsion, termination is no longer possible. Because the total concentration of radicals is distributed throughout the particles, the probability that any two radicals will meet and terminate is reduced. The ideal situation is one in which a lone radical in a particle can terminate only with a radical that enters the particle. This is known as the "zero-one" limit and in this case the rate of bi-molecular termination is controlled by radical entry alone.[76] This segregation could be exploited to boost the effectiveness of a controlled process by decreasing the bi-molecular termination events that lead to broader molecular weight distributions and therefore increasing overall rates of reaction. Butté et al.[58] predicted that in the absence of control agent partition, RAFT would benefit most from segregation. They suggest that ATRP and NMP are self-regulating in the sense that the mechanism forces the maintenance of an extremely small overall number of radicals in the system resulting therefore in low polymerization rates. It should be noted, however, at least with regard to NMP, that in actual practice partitioning often does occur and depending on the particulars of a given system any beneficial effects from segregation can be a complex function of temperature, thermal polymerization, solubility of the nitroxide and particle diameter.[17,49,77] In any event, the picture is clearer with respect to RAFT and in principle a thoughtful selection of components could ensure that the ability to achieve a "zero-one" state and accrue the benefits of segregation were a strong function of the particle size while affording the luxury of negligible influence of other factors. The second advantage obtains from a feature unique to the RAFT mechanism, in which very quickly the control agent becomes bound to a growing

polymer chain and remains attached throughout the reaction. The result is that the control agent is much more likely to remain within the polymer particle and not partition into the aqueous phase, making RAFT ideally suited for miniemulsion polymerization.

1.1.3 CRP in continuous reaction systems

Given a suitable CRP/dispersed media combination, there remains the issue of how best to employ that combination in an industrial setting. Continuous systems are attractive because of their high throughput and low operating and labor costs. Additionally, when combined with CRP, continuous processes also offer the potential to produce copolymers with more consistent composition as compared to batch or semi-batch systems. Continuous systems offer possibilities for polymer structure control that cannot be realized in batch systems. By combining stirred and tubular reactors in a train, one can in principle dictate the structure of the final copolymer. Such a reactor train could be used to design specific molecular structures to fit desired end-use properties. This allows the use of process design to carry out molecular design, or "product by process". To date, however, most of the reports in the published literature with regard to RAFT, ATRP and NMP have been conducted in batch or semi-batch reaction systems. In one notable exception, Zhu et al.[78,79] successfully employed continuous CRP of methyl methacrylate in a packed column reactor containing silica-supported atom transfer radical polymerization (ATRP) catalyst. Smulders et al.[62,63] have demonstrated the feasibility of employing RAFT in continuous stirred (CSTR) miniemulsion reactors to make both homo and copolymers. Enright et al.[80] employed NMP/miniemulsion in a tubular reactor

to make polystyrene. At present, there are no reports in the open literature employing ATRP/miniemulsion in a continuous reactor system.

1.1.4 CRP of vinyl acetate in miniemulsion

Poly(vinyl acetate), or PVA, is widely used in industry as a component in adhesives, paints and coatings. It is also important in its capacity as a precursor to poly(vinyl alcohol) or PVAL. PVAL is water soluble, non-toxic, non-carcinogenic and as such is of particular interest to the pharmaceutical industry. Among other things, it can be crosslinked to form a hydrogel which can be employed to deliver drugs diffusively. The ability to polymerize VA in a controlled manner so that highly defined PVAL architectures can be produced is therefore an attractive goal. As pointed out in the Section 1.1.1, of the three CRPs in common use, RAFT is generally considered the most versatile in terms of the range and functionality of monomers that can be polymerized. Section 1.1.2 emphasized another important advantage, that it can be used in aqueous systems at low temperatures. Even so, its use with vinyl esters, and in particular vinyl acetate (VA), has been limited. In the case of vinyl acetate, the propagating radical is poorly stabilized and therefore highly reactive. As such, it is a very poor leaving group. In terms of rate constants, this indicates that the addition rate constants will be much larger than the fragmentation rate constants in Scheme 1.1, **IV**. This in turn causes slow fragmentation of the intermediate radical, **5**. Polymerizations using standard RAFT dithioesters can exhibit complete inhibition.[81] While the dithioesters can be ineffective, xanthates and dithiocarbamates[81-88] have both been successfully utilized to produce poly(vinyl acetate)

in a controlled manner. Attempts to produce PVA homopolymer using NMP or ATRP have thus far met with little success.[89-91]

1.2 Summary

The goal of this work is to move forward the boundaries of knowledge in two areas related to CRPs. Progressing closer towards employing an aqueous system, specifically miniemulsion, to produce PVA via RAFT chemistry constitutes the first part of this goal. This work will be presented in Chapter 2. The second part of the goal is to further the understanding of the nature of the RAFT/miniemulsion system when employed in continuous tubular reactors. Chapter 3 will present the development of the recipe using mixed surfactants, the results of styrene homopolymerizations in batch and tube, and the results of a chain extension experiment demonstrating the living nature of the chains formed in the tubular reactor. Kinetic anomalies are addressed, as well as polydispersity differences between batch and tube. Flow phenomenon and their influence on residence time distribution and by implication the PDI of the polymer formed are the focus of Chapter 4. A model of RAFT in laminar flow is first presented and the results and implications are discussed. The flow profile of the reactor is examined using a tracer technique developed specifically for this system. Experiments are presented directly relating the residence time distribution to the polydispersity of the polymer. Transient behavior of the reactor in isolated plug flow is explained in terms of initiator loss. Both experimental data and a model are used to support this hypothesis. Finally, conclusions and implications, unanswered questions and the ideas for future work that they spawned are addressed in Chapter 5.

1.3 References

1. Recognizing that currently an ongoing debate in the scientific community revolves around whether to utilize "controlled" or "living" to properly characterize these mechanisms (see, for example, Darling T.R., et al., *J. Polym. Sci. Polym. Chem.* **2000**, *38*, 1706), in the context of this dissertation, they will be referred to as controlled. While they exhibit many of the characteristics of living polymerizations, they all come with inherent termination to one degree or another.

2. *Chem. Eng. News* **1997**, *75*, p 48.

3. Patten, T. E.; Matyjaszewski, K. *Adv. Mater.* **1998**, *10*, 901.

4. Kato, M.; Kamigaito, M.; Sawamoto, M.; Higashimura, T. *Macromolecules* **1995**, *28*, 1721.

5. Percec, V.; Barboiu, B. *Macromolecules* **1995**, *28*, 7970.

6. Wang, J. S.; Matyjaszewski, K. *J. Am. Chem. Soc.* **1995**, *117*, 5614.

7. Rizzardo, E. *Chem. Aust.* **1987**, *54*, 32.

8. Solomon, D. H.; Rizzardo, E.; Cacioli, P. U.S. 4581429, 1986.

9. Barner-Kowollik, C.; Quinn, J. F.; Nguyen, T. L. U.; Heuts, J. P. A.; Davis, T. P. *Macromolecules* **2001**, *34*, 7849.

10. Le, T. P. T.; Moad, C. L.; Rizzardo, E.; Thang, S. H. PCT Int. Appl. WO 9801478, 1998.

11. Rizzardo, E.; Thang, S. H.; Moad, G. PCT Int. Appl. WO 9905099, 1999.

12. Goto, A.; Sato, K.; Tsujii, Y.; Fukuda, T.; Moad, G.; Rizzardo, E.; Thang, S. H. *Macromolecules* **2001**, *34*, 402.

13. Wang, A. R.; Zhu, S. P. *J. Polym. Sci. Polym. Chem.* **2003**, *41*, 1553.

14. Chong, Y. K.; Krstina, J.; Le, T. P. T.; Moad, G.; Postma, A.; Rizzardo, E.; Thang, S. H. *Macromolecules* **2003**, *36*, 2256.

15. Matyjaszewski, K.; Xia, J. H. *Chem. Rev.* **2001**, *101*, 2921.

16. de Brouwer, H. RAFT Memorabilia: Living Radical Polymerization in Homogeneous and Heterogeneous Media. Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, 2001.

17. Smulders, W. Macromolecular Architecture in Aqueous Dispersions. Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, 2002.

18. Chiefari, J.; Mayadunne, R. T. A.; Moad, C. L.; Moad, G.; Rizzardo, E.; Postma, A.; Skidmore, M. A.; Thang, S. H. *Macromolecules* **2003**, *36*, 2273.

19. Luo, Y. W.; Tsavalas, J.; Schork, F. J. *Macromolecules* **2001**, *34*, 5501.

20. Ali, M. M.; Stoever, H. D. H. *Macromolecules* **2003**, *36*, 1793.

21. Percec, V.; Popov, A. V.; Ramirez-Castillo, E.; Monteiro, M.; Barboiu, B.; Weichold, O.; Asandei, A. D.; Mitchell, C. M. *J. Am. Chem. Soc.* **2002**, *124*, 4940.

22. Sarbu, T.; Pintauer, T.; McKenzie, B.; Matyjaszewski, K. *J. Polym. Sci. Polym. Chem.* **2002**, *40*, 3153.

23. Bicak, N.; Gazi, M.; Tunca, U.; Kucukkaya, I. *J. Polym. Sci. Polym. Chem.* **2004**, *42*, 1362.

24. De Clercq, B.; Verpoort, F. *Macromolecules* **2002**, *35*, 8943.

25. Jousset, S.; Qiu, J.; Matyjaszewski, K. *Macromolecules* **2001**, *34*, 6641.

26. Lenzi, M. K.; Cunningham, M. F.; Lima, E. L.; Pinto, J. C. *Ind. Eng. Chem. Res.* **2005**, *44*, 2568.

27. Wannemacher, T.; Braun, D.; Pfaendner, R. *Macromol. Symp.* **2003**, *202*, 11.

28. Zhu, C. Y.; Sun, F.; Zhang, M.; Jin, R. *Polymer* **2004**, *45*, 1141.

29. Marestin, C.; Noel, C.; Guyot, A.; Claverie, J. *Macromolecules* **1998**, *31*, 4041.

30. Cao, J. Z.; He, J. P.; Li, C. M.; Yang, Y. L. *Polymer Journal* **2001**, *33*, 75.

31. Delaittre, G.; Nicolas, J.; Lefay, C.; Save, M.; Charleux, B. *Chem. Commun.* **2005**, 614.

32. Nicolas, J.; Charleux, B.; Guerret, O.; Magnet, S. *Angew. Chem.* **2004**, *43*, 6186.

33. Prescott, S. W.; Ballard, M. J.; Rizzardo, E.; Gilbert, R. G. *Macromolecules* **2002**, *35*, 5417.

34. Smulders, W.; Gilbert, R. G.; Monteiro, M. J. *Macromolecules* **2003**, *36*, 4309.

35. Uzulina, I.; Kanagasabapathy, S.; Claverie, J. *Macromol. Symp.* **2000**, *150*, 33.

36. Prescott, S. W.; Ballard, M. J.; Rizzardo, E.; Gilbert, R. G. *Macromolecules* **2005**, *38*, 4901.

37. Apostolovic, B.; Quattrini, F.; Butté, A.; Storti, G.; Morbidelli, M. *DECHEMA Monographien* **2004**, *138*, 115.

38. Monteiro, M. J.; Hodgson, M.; De Brouwer, H. *J. Polym. Sci. Polym. Chem.* **2000**, *38*, 3864.

39. Monteiro, M. J.; Sjoberg, M.; van der Vlist, J.; Gottgens, C. M. *J. Polym. Sci. Polym. Chem.* **2000**, *38*, 4206.

40. Tsavalas, J. G.; Schork, F. J.; de Brouwer, H.; Monteiro, M. J. *Macromolecules* **2001**, *34*, 3938.

41. de Brouwer, H.; Tsavalas, J. G.; Schork, F. J.; Monteiro, M. J. *Macromolecules* **2000**, *33*, 9239.

42. Landfester, K.; Bechthold, N.; Tiarks, F.; Antonietti, M. *Macromolecules* **1999**, *32*, 2679.

43. Chern, C. S.; Liou, Y. C. *Macromol. Chem. Physic.* **1998**, *199*, 2051.

44. Moad, G.; Chiefari, J.; Chong, Y. K.; Krstina, J.; Mayadunne, R. T. A.; Postma, A.; Rizzardo, E.; Thang, S. H. *Polym. Int.* **2000**, *49*, 993.

45. Monteiro, M. J.; de Barbeyrac, J. *Macromolecules* **2001**, *34*, 4416.

46. Butté, A.; Quattrini, F.; Storti, G.; Morbidelli, M. EP 1533327, 2005.

47. MacLeod, P. J.; Barber, R.; Odell, P. G.; Keoshkerian, B.; Georges, M. K. *Macromol. Symp.* **2000**, *155*, 31.

48. Keoshkerian, B.; MacLeod, P. J.; Georges, M. K. *Macromolecules* **2001**, *34*, 3594.

49. Cunningham, M. F.; Tortosa, K.; Ma, J. W.; McAuley, K. B.; Keoshkerian, B.; Georges, M. K. *Macromol. Symp.* **2002**, *182*, 273.

50. Cunningham, M. F.; Xie, M.; McAuley, K. B.; Keoshkerian, B.; Georges, M. K. *Macromolecules* **2002**, *35*, 59.

51. Li, M.; Matyjaszewski, K. *J. Polym. Sci. Polym. Chem.* **2003**, *41*, 3606.

52. Li, M.; Jahed, N. M.; Min, K.; Matyjaszewski, K. *Macromolecules* **2004**, *37*, 2434.

53. Li, M.; Min, K.; Matyjaszewski, K. *Macromolecules* **2004**, *37*, 2106.

54. Min, K. E.; Li, M.; Matyjaszewski, K. *J. Polym. Sci. Polym. Chem.* **2005**, *43*, 3616.

55. Kagawa, Y.; Minami, H.; Okubo, M.; Zhou, J. *Polymer* **2005**, *46*, 1045.

56. Li, M.; Matyjaszewski, K. *Macromolecules* **2003**, *36*, 6028.

57. Li, C.; Min, K.; Matyjaszewski, K. *Macromolecules* **2004**.

58. Butté, A.; Storti, G.; Morbidelli, M. *Macromolecules* **2001**, *34*, 5885.

59. Lansalot, M.; Davis, T. P.; Heuts, J. P. A. *Macromolecules* **2002**, *35*, 7582.

60. Mcleary, J. B.; Tonge, M. P.; Roos, D. D.; Sanderson, R. D.; Klumperman, B. *J. Polym. Sci. Pol. Chem.* **2004**, *42*, 960.

61. Luo, Y. W.; Liu, X. *J. Polym. Sci. Pol. Chem.* **2004**, *42*, 6248.

62. Smulders, W. W.; Jones, C. W.; Schork, F. J. *AIChE J.* **2005**, *51*, 1009.

63. Smulders, W. W.; Jones, C. W.; Schork, F. J. *Macromolecules* **2004**, *37*, 9345.

64. Russum, J. P.; Jones, C. W.; Schork, F. J. *Macromol. Rapid Comm.* **2004**, *25*, 1064.

65. Russum, J. P.; Jones, C. W.; Schork, F. J. *Ind. Eng. Chem. Res.* **2005**, *44*, 2484.

66. Russum, J. P.; Barbre, N. D.; Jones, C. W.; Schork, F. J. *J. Polym. Sci. Polym. Chem.* **2005**, *43*, 2188.

67. Simms, R. W.; Davis, T. P.; Cunningham, M. F. *Macromol. Rapid Comm.* **2005**, *26*, 592.

68. Luo, Y.; Yang, L.; Li, B. *DECHEMA Monographien* **2004**, *138*, 473.

69. Ugelstad, J.; El-Aasser, M. S.; Vanderhoff, J. W. *J. Polym. Sci. Pol. Lett.* **1973**, *11*, 503.

70. Gromada, J.; Matyjaszewski, K. *Macromolecules* **2001**, *34*, 7664.

71. Gaynor, S. G.; Qiu, J.; Matyjaszewski, K. *Macromolecules* **1998**, *31*, 5951.

72. Tortosa, K.; Smith, J. A.; Cunningham, M. F. *Macromol. Rapid Comm.* **2001**, *22*, 957.

73. Braslau, R.; O'Bryan, G.; Nilsen, A.; Henise, J.; Thongpaisanwong, T.; Murphy, E.; Mueller, L.; Ruehl, J. *Synthesis-Stuttgart* **2005**, 1496.

74. Drockenmuller, E.; Lamps, J. P.; Catala, J. M. *Macromolecules* **2004**, *37*, 2076.

75. Moad, G.; Mayadunne, R. T. A.; Rizzardo, E.; Skidmore, M.; Thang, S. H. *Macromol. Symp.* **2003**, *192*, 1.

76. Gilbert, R. G., *Emulsion Polymerization : A Mechanistic Approach*. Academic Press: 1995.

77. Charleux, B. *Macromolecules* **2000**, *33*, 5358.

78. Shen, Y. Q.; Zhu, S. P.; Pelton, R. *Macromol. Rapid Comm.* **2000**, *21*, 956.

79. Shen, Y. Q.; Zhu, S. P. *AIChE J.* **2002**, *48*, 2609.

80. Enright, T. E.; Cunningham, M. F.; Keoshkerian, B. *Macromol. Rapid Comm.* **2005**, *26*, 221.

81. Rizzardo, E.; Chiefari, J.; Mayadunne, R. T. A.; Moad, G.; Thang, S. H. *ACS Symp. Ser.* **2000**, *768*, 278.

82. Destarac, M.; Charmot, D.; Franck, X.; Zard, S. Z. *Macromol. Rapid Comm.* **2000**, *21*, 1035.

83. Stenzel, M. H.; Cummins, L.; Roberts, G. E.; Davis, T. R.; Vana, P.; Barner-Kowollik, C. *Macromol. Chem. Physic.* **2003**, *204*, 1160.

84. Coote, M. L.; Radom, L. *Macromolecules* **2004**, *37*, 590.

85. Favier, A.; Barner-Kowollik, C.; Davis, T. P.; Stenzel, M. H. *Macromol. Chem. Physic.* **2004**, *205*, 925.

86. Stenzel, M. H.; Davis, T. P.; Barner-Kowollik, C. *Chem. Commun.* **2004**, 1546.

87. Destarac, M.; Taton, D.; Zard, S. Z.; Saleh, T.; Yvan, S. *ACS Symp. Ser.* **2003**, *854*, 536.

88. Charmot, D.; Corpart, P.; Adam, H.; Zard, S. Z.; Biadatti, T.; Bouhadir, G. *Macromol. Symp.* **2000**, *150*, 23.

89. Simal, F.; Delfosse, S.; Demonceau, A.; Noels, A. F.; Denk, K.; Kohl, F. I.; Weskamp, T.; Herrmann, W. A. *Chem.-Eur. J.* **2002**, *8*, 3047.

90. Wakioka, M.; Baek, K. Y.; Ando, T.; Kamigaito, M.; Sawamoto, M. *Macromolecules* **2002**, *35*, 330.

91. Xia, J. H.; Paik, H. J.; Matyjaszewski, K. *Macromolecules* **1999**, *32*, 8310.

CHAPTER 2

POLYMERIZATION OF VINYL ACETATE USING RAFT IN MINIEMULSION[†]

2.1 Introduction

As pointed out in the first chapter, of the three CRPs in common use, RAFT is generally considered the most versatile in terms of the range and functionality of monomers that can be polymerized. Another important advantage is that it can be used in aqueous systems at low temperatures. Even so, its use with vinyl esters, and in particular vinyl acetate (VA), has been limited. In the case of vinyl acetate, the propagating radical is very unstable and highly reactive. Polymerizations using standard RAFT dithioesters can exhibit complete inhibition.[1] It has been postulated that the cause is slow fragmentation of the RAFT intermediate radical and is brought about by the fact that the vinyl acetate propagating radical is such a poor leaving group.[2] While the dithioesters can be ineffective, xanthates and dithiocarbamates[1-8] have both been successfully utilized to produce poly(vinyl acetate) in a controlled manner. Attempts to produce PVA homopolymer using NMP or ATRP have thus far met with little success.[9-11] Poly(vinyl acetate), or PVA, is widely used in industry as a component in adhesives, paints and coatings. It is also important in its capacity as a precursor to poly(vinyl alcohol) or PVAL. PVAL is water soluble, non-toxic, non-carcinogenic and as such is of particular interest to the pharmaceutical industry. Among other things, it can be crosslinked to form a hydrogel which can be employed to deliver drugs diffusively. The ability to polymerize

[†] Portions of this chapter have been previously published, *J. Polym. Sci. Polym. Chem.* **2005**, *43*, 2188.

VA in a controlled manner so that highly defined PVAL architectures can be produced is therefore an attractive goal.

As already mentioned, several studies in the open literature have reported controlled polymerizations of VA using RAFT agents. These studies utilized either bulk or solution techniques, but the commercial viability of VA/RAFT would be greatly enhanced if it could be shown to be successful in an aqueous system. Herein is presented the successful miniemulsion polymerization[12] of VA using methyl (ethyloxycarbonothioyl)sulfanyl acetate as the RAFT agent (MESA, Scheme 2.5).[§] In a study of eight xanthates by Stenzel, et al.,[2] MESA was shown to provide the lowest polydispersities in combination with the highest rates of polymerization in bulk VA polymerizations, up to conversions of ~ 60%. Because VA is relatively soluble in water (~ 2.8 wt% at 60 ºC)[13], an oil soluble initiator, 2,2' azo-bis(isobutyronitrile) (AIBN), was initially utilized to prevent nucleation in the aqueous phase and suppress the formation of uncontrolled, free radical polymer. Simms et al.[14] have also reported the miniemulsion polymerization of VA using RAFT and a water soluble initiator, 2,2'-azobis[2-(2-dimidazolin-2-yl)propane] dihydrochloride (VA-044). While the polymerizations exhibited control, the molecular weights were higher than theoretical, polydispersities at high conversions were higher than typically seen in controlled polymerizations (~1.6-1.9) and limiting conversions were observed.

2.2 Results and discussion

Initially, six reactions were conducted as outlined in Table 2.1. AIBN was used as initiator because of concerns about nucleation in the aqueous phase and the subsequent

[§] The rationale for employing miniemulsion is outlined in Chapter 1.

formation of uncontrolled free-radical PVA. Vinyl acetate is a relatively water soluble monomer, so it is possible that the use of a water soluble initiator such as potassium persulfate (KPS) could trigger the formation of propagating monomeric radicals in the aqueous phase. These radicals would not have access to the relatively insoluble MESA and would tend to form uncontrolled, high molecular weight polymer via polymerization

Table 2.1 Summary of the Experimental Conditions of the Polymerizations Shown in This Study

Exp	Type	$[MESA]_0/[AIBN]_0$	Temp, ºC
1	Bulk	5.0	60
2	Mini	5.0	60
3	Mini	2.5	60
4	Mini	1.0	60
5	Mini	2.5	70
6	Mini	1.0	70

in aqueous phase. This would be reflected in the molecular weight distribution as a bimodality, making it difficult to determine the "true" molecular weight of the controlled polymer. Additionally, because the aqueous polymerization would deprive the controlled polymerization (taking place in the droplets/particles) of monomer, the number average weight of the controlled polymer would be artificially depressed and difficult to control to a reasonable degree of accuracy.

Figure 2.1 shows the kinetic results of the six experiments. The salient aspect to note is the marked difference between the kinetics of the bulk polymerization (Exp 1) and the same recipe in terms of AIBN and MESA concentration in miniemulsion (Exp 2, see

Table 2.1). Under normal circumstances, it would be expected that kinetically the miniemulsion would produce faster kinetics than bulk, owing to the beneficial effects of segregation.[15,16] In this case, however, the same recipe polymerized in miniemulsion was much slower than the bulk experiment. While the kinetics were improved by increasing the temperature and/or the starting initiator concentration (Exp 3-6), in each case the conversion stops short of that achieved in bulk. The highest conversion in this

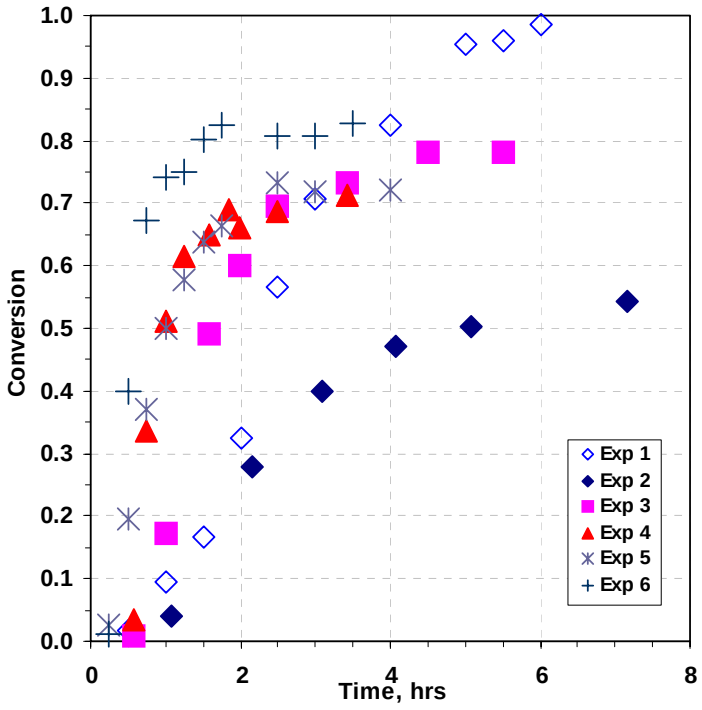


Figure 2.1 Kinetic plots of bulk and miniemulsion polymerizations conducted in this study.

group of miniemulsion polymerizations was ~ 83%, whereas the conversion reached almost 100% in bulk at a lower temperature (60 °C vs. 70 °C) and ~ 5 times less AIBN. Figure 2.2, comparing the molecular weight evolution and polysdispersity of the bulk

experiment and two of the higher conversion experiments shows that the experiments were largely controlled, however the final polydispersities were higher (~1.4-.16) than usually observed in polymerizations of this type. An examination of the GPC chromatograms reveals little or no high polymer that would be indicative of aqueous phase nucleation. A representative example (Exp 5) is shown in Figure 2.3. It shows the

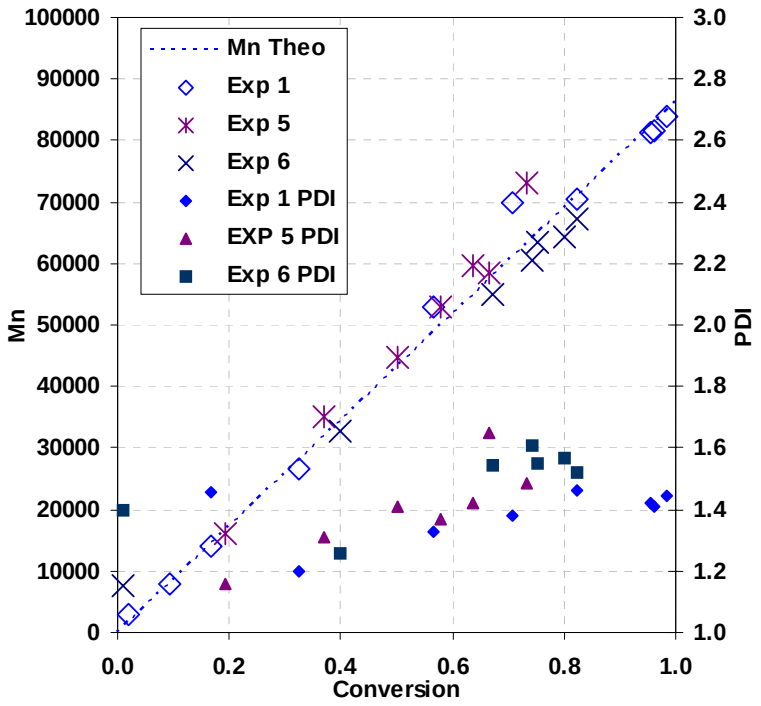


Figure 2.2 Evolution of Mn and PDI as a function of monomer conversion for selected experiments.

shift towards higher molecular in the chromatograms that is characteristic of controlled polymerizations, but there is no evidence of a population of uncontrolled high molecular weight polymer. This is critically important because the presence of high molecular

weight, uncontrolled polymer would make this method much less attractive as a means of producing controlled PVA in an industrial setting.

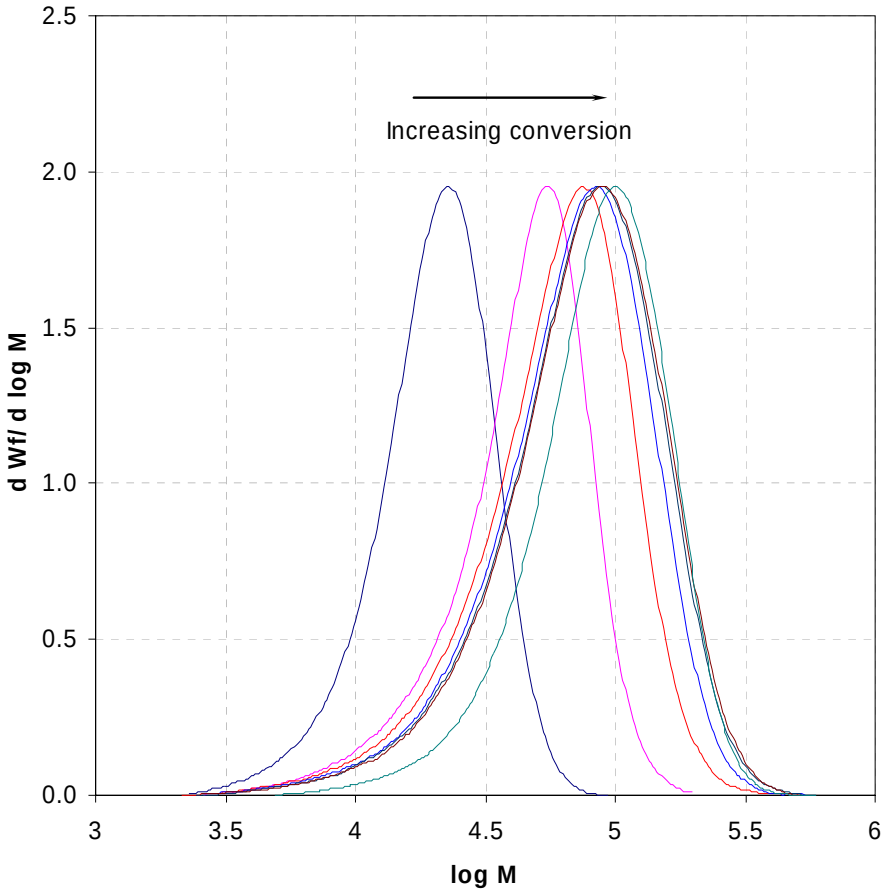


Figure 2.3 Evolution of molar mass distribution (Exp 5) for MESA mediated vinyl acetate miniemulsion polymerization. Curves represent monomer conversions of 19%, 37%, 50%, 58%, 64%, 66%, 73%, respectively.

2.2.1 Observed limiting conversion

At this point, the most important question becomes how to account for the limiting conversion and the discrepancy between miniemulsion and bulk kinetics. The possibilities that will be examined in further detail are:

1. Glass effect or gel formation
2. Hydrolysis of RAFT agent
3. Solubility of vinyl acetate and/or MESA RAFT agent contributing to radical desorption
4. Complications arising from use of oil-soluble initiator, either by irreversible termination of initiator radicals before they can form a growing chain or desorption of radicals to the aqueous phase

2.2.2 Glass effect, gel formation

It is well established that glass effects and gel formation can contribute to kinetic anomalies and limiting conversions in dispersed aqueous systems. In miniemulsions, the glass effect can arise when a polymer with a T_g higher than the polymerization temperature forms a shell around the particle that impedes the entry of radicals from the outside, effectively quenching the reaction.[17] For example, poly(methyl methacrylate) has a T_g of around 105 °C, a temperature obviously higher than would be used aqueous dispersions run at atmospheric pressure. However this system would not likely be affected by this because the T_g of PVA is ~ 32 ºC [18] and the polymerizations were performed at 60 or 70 ºC. What is more common in VA polymerizations is gel formation

because of branching. Vinyl acetate undergoes considerable chain transfer to polymer, forming branches.[19,20] Under certain circumstances this leads to gelation which can inhibit the access of monomer to the reaction locus. A well established measure of branching is the hydrodynamic volume of the molecule. Because the volume of branched polymer molecules is smaller than that of linear molecules having the same molecular weight[21] , the intrinsic viscosity of the branched polymer will be lower than a linear polymer of the same molecular weight. As such, the branched polymer can be identified using GPC in combination with a viscosity detector and laser light scattering.[22-24] Since this is an absolute characterization technique, conformational plots of log R_{RMS} vs. log M can be generated. The slope of the plot indicates whether or not branching is present. Linear polymers are expected to have a slope between ~ 0.5-0.6[24] and slopes of less than 0.5 imply branching. While these values may vary, what will always be the case is that for the same polymer species (e.g., PVA) the linear polymer will have a steeper slope than the branched polymer. Figure 2.4 compares two samples of the controlled PVA, one from the bulk polymerization and one from a miniemulsion polymerization (Exps 1 & 2), to a sample of branched and linear PVA. The slopes of the three linear samples are essentially the same, ~ 0.56, indicating that neither of the controlled samples is branched to any significant degree, effectively ruling this out as a source of the limiting conversion.

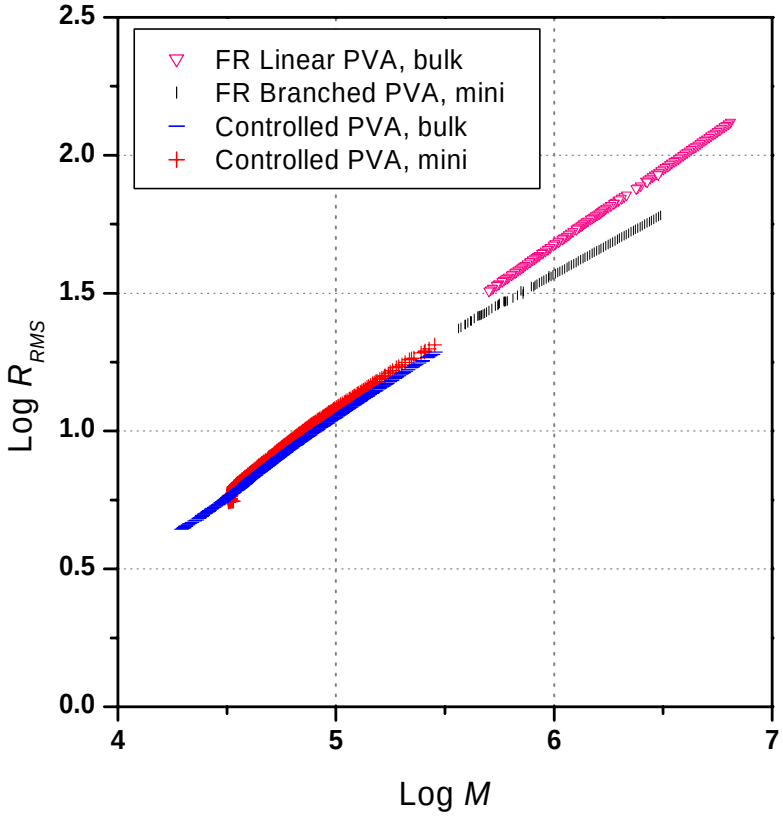


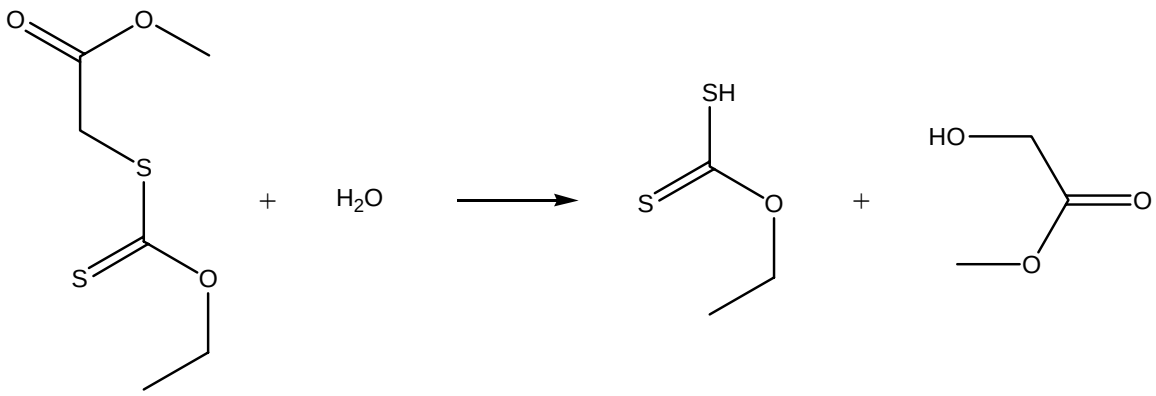
Figure 2.4 Log R_{RMS} vs. Log M: Samples from miniemulsion and bulk controlled PVA at ~ 55% conversion. Compared to branched and linear free radical PVA. Slopes of controlled PVA are essentially the same as that of linear PVA (~ 0.56, typically associated with linear PVA), no indication of higher branching in miniemulsion. Data obtained via SEC-Vis-RALLS.

2.2.3 Hydrolysis/decomposition of the RAFT agent

Another possibility that was considered is hydrolysis of the MESA RAFT agent. Dithioesters have been reported to hydrolyze under certain temperature and pH conditions.[25-27] While the loss of the RAFT agents' ability to mediate the polymerization

could certainly lead to limiting conversions, what was not observed in this series of experiments was an increase in the number average molecular weight that would also result from significant loss of MESA. In each case the actual number average molecular weights were close to theoretical (see Figure 2.2). Temperature and pH both play a significant role in the degree to which hydrolysis occurs, higher temperatures and higher pH (varies, but generally > 7) tend to promote hydrolysis of dithioesters.[26-29] The temperatures of the reactions here could be favorable, but miniemulsions using anionic surfactants (the pH of a 1% solution of SDS is 6) lean towards the acidic which would tend to suppress hydrolysis.

To address this question fully, a series of experiments was conducted to monitor the MESA and determine if there was any formation of hydrolysis products or loss of the C=S functionality. Using the same proportions as the miniemulsion recipe (see Table 2.2), the MESA was combined with water and two different combinations of the remaining components as outlined in Table 2.3. This rationale was to combine the reactants in such a way as to avoid polymerization but not the possibility of hydrolysis or some other reaction that might decompose the RAFT agent. The mixtures were then "cooked" and sampled in the same manner as the miniemulsion experiments (see Section 2.4.6), then the dried samples were dissolved in THF and injected in the GPC and the RI and UV signals recorded.



1 **2** **3**

Scheme 2.1 Hydrolysis reaction of dithioester RAFT agent

Scheme 2.1 shows the hydrolysis reaction for MESA (**1**), forming ethylxanthic acid (**2**) and methyl glycolate (**3**). Since in this case the C=S double bond remains intact, any formation of ethylxanthic acid would show up as a peak towards lower molecular weight in the SEC/UV chromatogram. Comparison to the UV chromatogram of pure MESA would indicate the formation of **2**. However as Figure 2.5 shows, when the UV chromatograms of the three experiments are compared to pure MESA, very little difference is observed. The peak retention volumes are essentially identical and there is no evidence in any of the three experiments that a lower molecular weight species of any kind has formed that includes the C=S bond. This effectively rules out hydrolysis of the MESA RAFT agent as a contributing factor in the limiting conversion. Interestingly, a small amount of some higher molecular weight material is indicated in the chromatogram from DCMP 3. Given that this experiment included the initiator AIBN,

this may represent oligomeric RAFT agent formed from the decomposition of AIBN. As

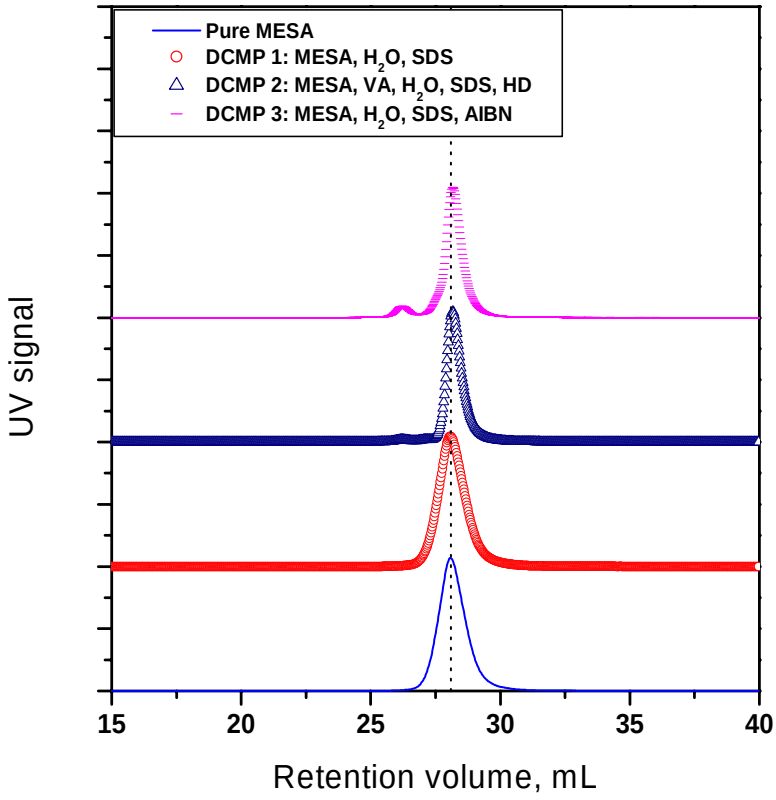
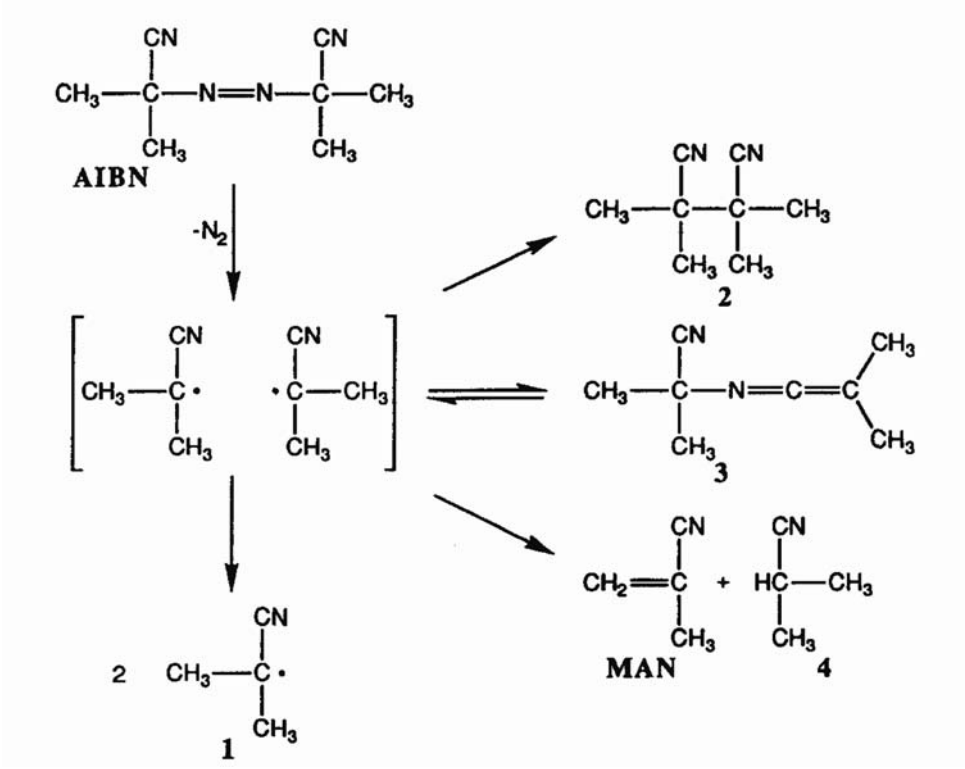


Figure 2.5 Comparison of SEC/UV chromatograms of MESA



Scheme 2.2 Decomposition products of AIBN. Adapted from [30].

shown in Scheme 2.2, one of the products of AIBN decomposition is methacrylonitrile (MAN).[30-37] While the amount formed is very small, ~ 3%,[30] it has been shown to oligomerize[30,37] with the cyanoisopropyl radical **1**. However, in the presence of a RAFT agent it would take on a similar role as the monomer in the pre-equilibrium phase (see Scheme 1.1, **II**) of the reaction, forming oligomeric RAFT molecules. These would maintain the C=S bond and therefore be reflected in the SEC/UV chromatogram as separate peaks at slightly higher molecular weights (i.e., lower retention volumes) than the pure MESA RAFT agent. This fits well with the chromatograms shown in Figure 2.6, which compares the first and last samples of DCMP 3. The samples were taken at 30

minutes and 4 hours, respectively. The first sample clearly shows the formation of at least two small populations of some higher molecular weight material with a C=S bond. As the

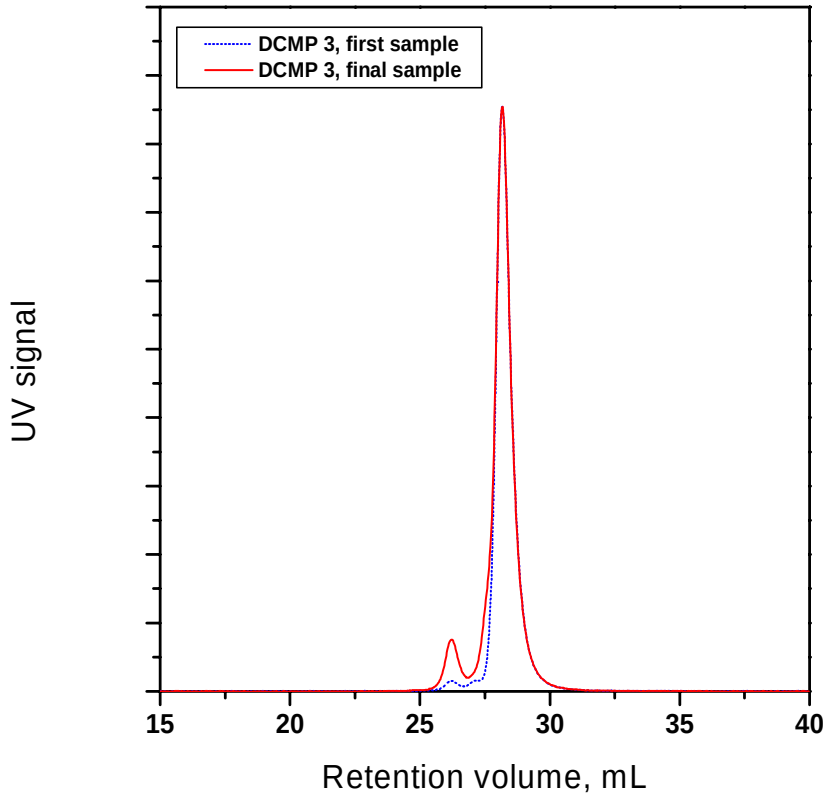


Figure 2.6 Comparison of SEC/UV chromatograms of DCMP 3, first and final samples

reaction progresses the populations become larger, possibly overlapping into the peak observed to the left in the final sample. While not conclusive, it is consistent with what would be observed provided RAFT oligomers were forming. Based on these observations, it is unlikely that hydrolysis is occurring to any degree that would impact negatively on the polymerization.

To assess whether some other reaction might be occurring that degraded the functionality of the MESA, the areas under the RI and UV curves of the chromatograms were compared. A relationship between the total mass of reactants and the total number of C=S bonds remaining can be established by simply taking the ratio of the areas under the peaks. A significant decrease in the UV/RI ratio would point towards loss of the C=S bond because the total mass (RI area) would not change while the number of C=S bonds (UV area) would decrease. Figure 2.7 shows the UV/RI area normalized ratios plotted against time for the three experiments. At worst, they show a slight downward (i.e. loss of C=S) but inconsequential trend in the ratios. This suggests that it is doubtful that the MESA decomposes to a degree large enough to affect the polymerization. Taken in tandem with the results shown in Figures 2.5 and 2.6, both hydrolysis and/or some other decomposition of the C=S bond are ruled out as causes for the observed limiting conversions.

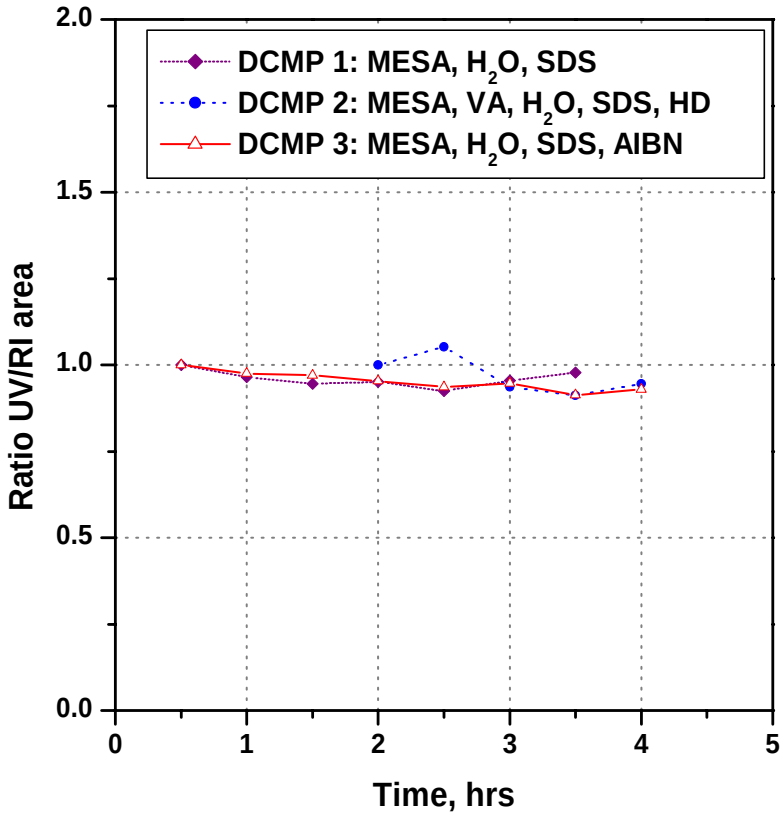
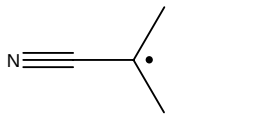


Figure 2.7 Plot of the normalized ratios of the area of the UV/RI chromatograms for DCMP 1-3.

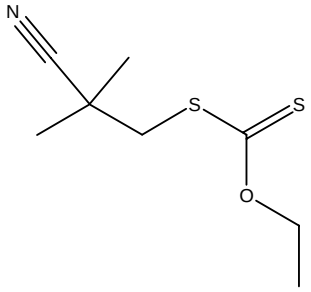
2.2.4 Desorption of reactants

Reports in the open literature of the miniemulsion homopolymerizations of vinyl acetate using conventional free radical polymerization (FRP) make no mention of limiting conversions,[38-43] although they have been observed in suspension polymerizations.[44,45] A characteristic that distinguishes CRPs from FRPs is the period of time early in the reaction when only small oligomers exist, whether dormant or propagating. It is generally accepted that this attribute is primarily responsible for the latex instability that is often observed in CRP/miniemulsion systems[46]. While latex instability is a thermodynamic issue, this particular aspect could certainly have an influence on kinetic phenomena as well. Radical desorption, droplet nucleation, and termination could all be impacted. Because of the water solubility of vinyl acetate (~ 2.8 wt% at 60 ºC)[13], relatively high radical desorption can occur more readily than with a more hydrophobic monomer, such as styrene for example. Another level of uncertainty in the VA/MESA system is the solubility of the RAFT agent. Scheme 2.3 shows several of the species present during the early stages of the polymerization that could be susceptible to desorption. While they all would have some effect upon the polymerization should they escape the droplet, the two most important in terms of the observed limiting conversion are likely **D** and **E**. If they desorb in large enough numbers and then terminate before re-entering another droplet, they would effectively rob the polymerization of monomer.

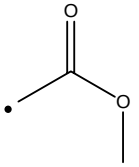
To test whether either of these might play a role in the observed discrepancy between bulk and miniemulsion kinetics, a seeded miniemulsion was conducted. The MESA



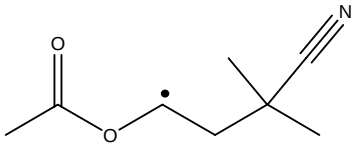
A



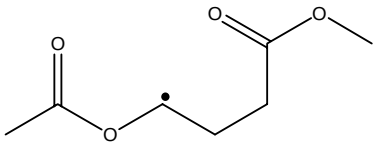
B



C



D



E

Scheme 2.3 Primary species of interest during the early stages of VA/MESA polymerization. **A**: cyanoisopropyl radical, **B**: product of **A** and MESA, **C**: radical formed from R-group of MESA, **D**: radical formed from **A** and VA, **E**: radical formed from **C** and VA

RAFT agent was modified to incorporate an octyl tail on the Z-group (MOSA, Scheme 2.5). This would in principle impart a much greater degree of hydrophobicity to the RAFT agent without drastically altering the activity. As such, the amount of the MOSA analog to Scheme 2.3 **B** that escaped would be exceedingly small. By first polymerizing in bulk and creating oligomers, species **C** and **E** can be greatly reduced or eliminated. This is because each of the R-group derived radicals **C** initiates a chain **E** that propagates to some larger degree of polymerization. Each **C** will now be at the end of one of those chains and no more **E** will be formed. Because these chains are larger, their partition is unlikely. Of the five, this leaves **A** and **D** as the only species that may exit the droplet and terminate. Since the concentration of **A** would not be affected, unless **D** is relatively water soluble and **E** relatively insoluble, an observable difference in conversion should be seen when compared to the "unseeded" MESA system if desorption of **B**, **C**, **D** or **E** is an issue.

A bulk experiment was first conducted to determine whether or not the MOSA exhibited comparable reactivity to the MESA. Figure 2.8 shows the results compared to the bulk MESA experiment presented earlier. The actual molecular weights agree well with predicted and the polydispersity was similar to those achieved using MESA. Kinetically the MOSA polymerization was slightly retarded when compared to the MESA. Since the recipe parameters were the same, the source could be the difference in structure of the two RAFT agents. Xanthates are effective with VA in part because they

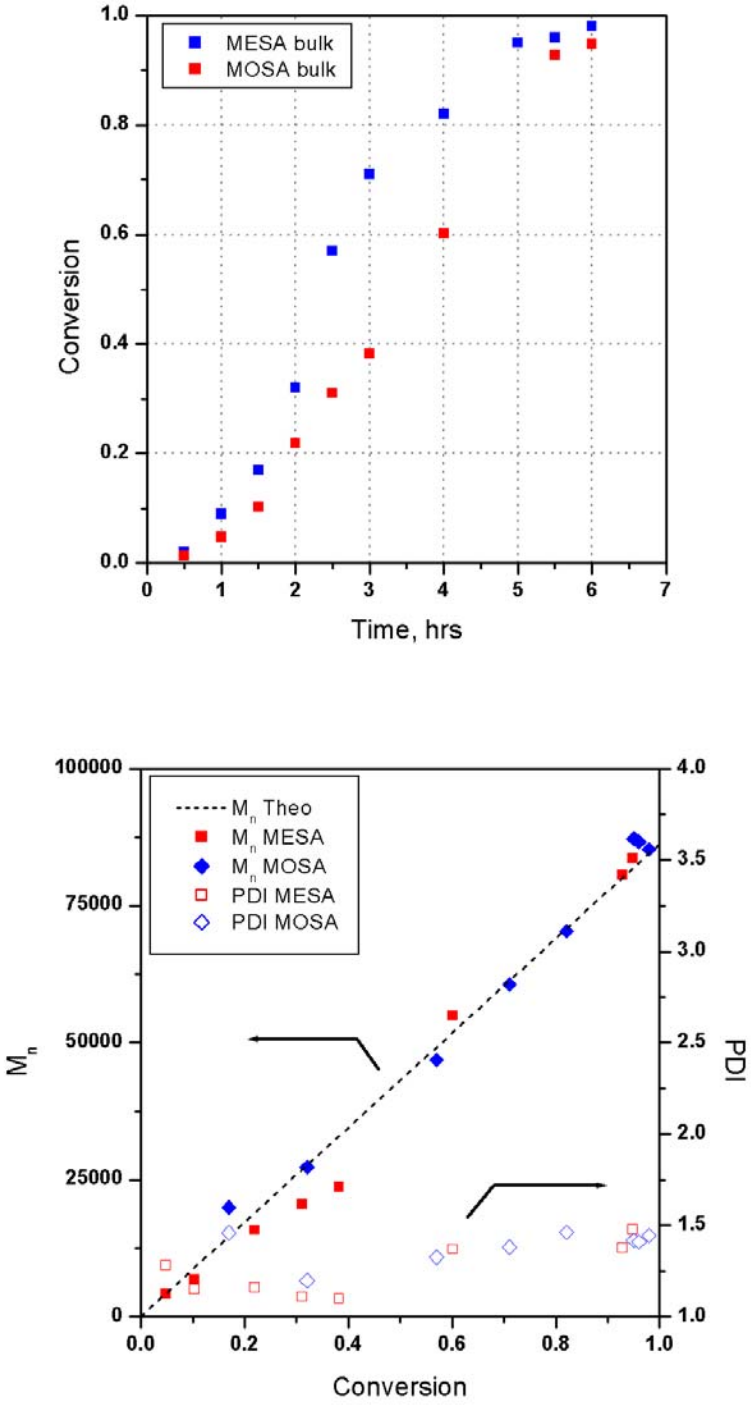
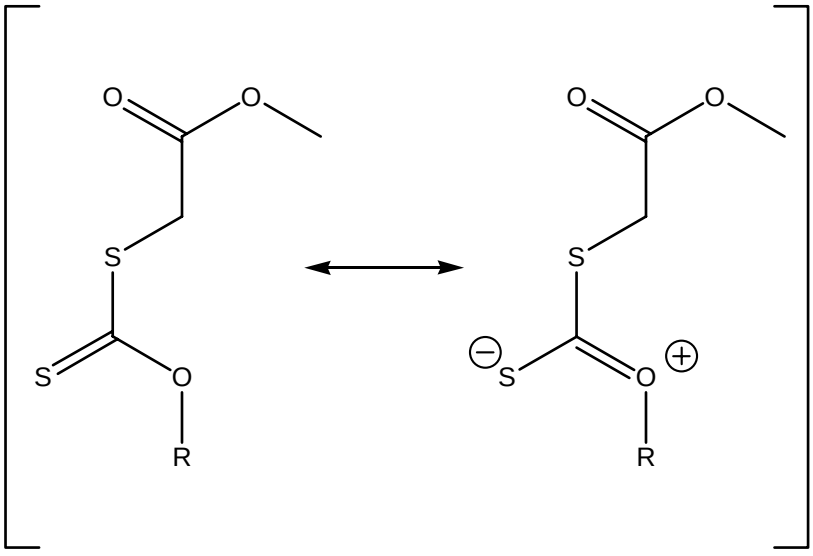


Figure 2.8 Comparison of vinyl acetate bulk polymerizations with MESA and MOSA. $[VA]_0/[AIBN]_0 \sim 5000$: $[VA]_0/[MOSA]_0 \sim 1000$: $[MOSA]_0/[AIBN]_0 \sim 5$

decrease the double bond character of the C=S bond (see Scheme 2.4), increasing the likelihood of fragmentation of the intermediate RAFT radical (Scheme 1.1, **3** and **5**). The ^{1}H NMR shifts of the protons in the methylene group adjacent to the sulfur suggest that the Z-group of the MOSA (~3.90 ppm) has slightly less electron withdrawing capacity than the ethoxy Z-group of the MESA (~3.92 ppm). This could in principle slow down the rate of fragmentation and tend to retard the polymerization.



Scheme 2.4 Proposed resonance structure for the two xanthate RAFT agents used in the study.

In any event, the difference is not significant enough to invalidate a comparison to the seeded experiment. However, as the results in Figure 2.9 show, the seeded polymerization using MOSA also reaches a limiting conversion, although slightly lower than the recipe using MESA without pre-polymerization. This strongly suggests that the

limiting conversion is not related to desorption of the species **B**, **C**, **D** and **E**. If correct, then the results imply that the culprit may be **A** since its fate is not substantially affected by the seeded polymerization. While the concentration of **A** would be the same in both the seeded and unseeded miniemulsion, other complications arising from the use of an oil-soluble initiator could come into play. These are addressed in the following section.

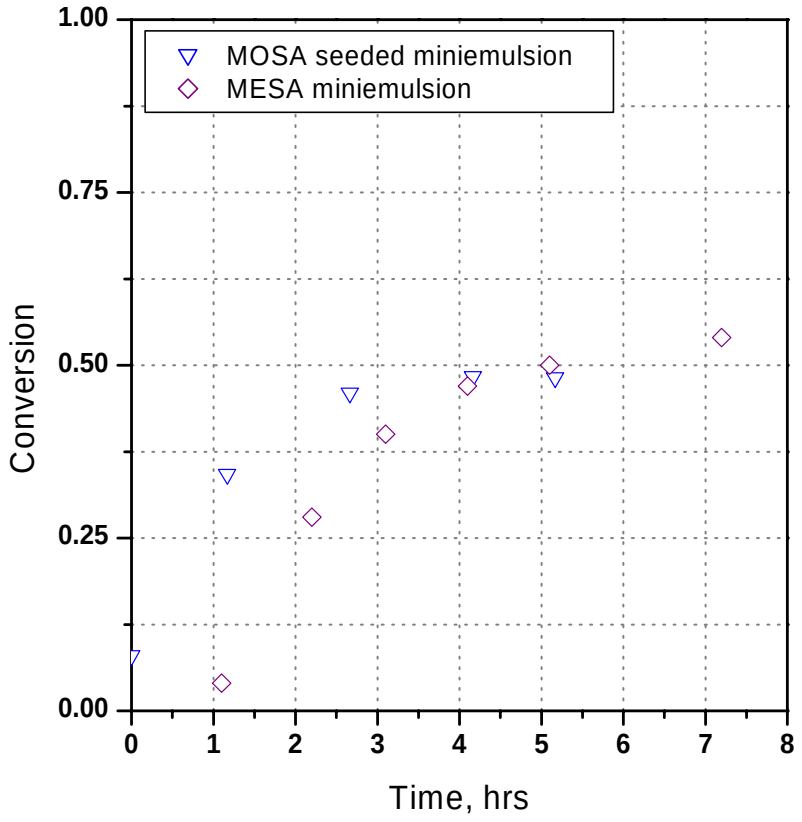


Figure 2.9 Plot of conversion vs. time VA/MOSA in seeded miniemulsion using macro-RAFT. $[VA]_0/[AIBN]_0$ ~ 5000 : $[VA]_0/[MOSA]_0$ ~ 1000 : $[MOSA]_0/[AIBN]_0$ ~ 5.

2.2.5 Complications arising from use of oil-soluble initiator

While an oil soluble initiator could greatly reduce the possibility of aqueous phase nucleation, their use in miniemulsions has been limited historically because of complications that can take place which adversely affect the polymerization.[47-61] These complications are thus far not fully understood and the exact mechanism has yet to be positively identified, but two most prominent hypotheses regarding the mechanism of radical generation can be categorized as follows:

1. The appearance of single radicals either by desorption of one of the radicals formed in the droplet/particle by initiator decomposition or by entry of a radical from the aqueous phase. [49,50]

2. All radicals formed from decomposition of the initiator that is present in the droplet/particle terminate before propagation. The kinetics are controlled by entry of single radicals formed by decomposition of the fraction of oil-soluble initiator dissolved in the aqueous phase.[51-54]

The first hypothesis maintains that when an initiator molecule decomposes, the radicals very quickly desorb from the droplet before they can initiate a polymer chain or terminate with another radical. They go on to enter into most likely a different droplet, and so long as there is not another radical (a "zero-one" state) they will go on to form a polymer chain. In this case the number of radicals is determined by the number of radicals that desorb as well as those that form from that fraction of initiator molecules that partition to the aqueous phase. The second hypothesis contends that because of the constrained volumes associated with miniemulsion droplets, the majority of the initiator dissolved in the organic phase will experience geminate, or instantaneous, termination

upon decomposition.[62] As such, the number of radicals is determined entirely by the initiator dissolved in the aqueous phase. The less water soluble an initiator is, the fewer the molecules that will be present in the aqueous phase to form radicals. Proponents of both theories have presented supporting evidence and settling the debate is beyond the scope of this work. Suffice it to say that either of the two will tend to reduce the total number of available initiator radicals that would otherwise be available to the system, effectively lowering the radical concentration as compared to the same starting concentration of initiator in a bulk system. This can be most clearly observed in the work of Choi et al.[47] where styrene polymerizations were conducted in miniemulsion using both water soluble (KPS) and oil-soluble (2,2´-azobis-(2-methyl butyronitrile), AMBN). At the reaction temperature of 70°C the half-life of both of the initiators is almost identical,[18] but the AMBN initiated polymerizations exhibited slower rates when compared to the KPS initiated polymerizations using an otherwise identical recipe in terms of the amounts of styrene, water, surfactant and costabilizer. This occurred even though in each case the AMBN concentrations were slightly higher.

To examine whether this plays a role here, two polymerizations were carried out using KPS as initiator. The KPS concentrations employed were the same as Exp 2 and Exp 3 using AIBN (see Table 2.1), where $[MESA]_0/[I]_0$ is 5.0 and 2.5, respectively, and nothing else was altered. When the kinetics of the KPS polymerizations are compared to the kinetics of the original recipes using the AIBN, the difference is quite notable (Figure 2.10). In both cases the KPS polymerizations achieved significantly higher conversions than the AIBN initiated polymerizations. However, in both instances the KPS initiated

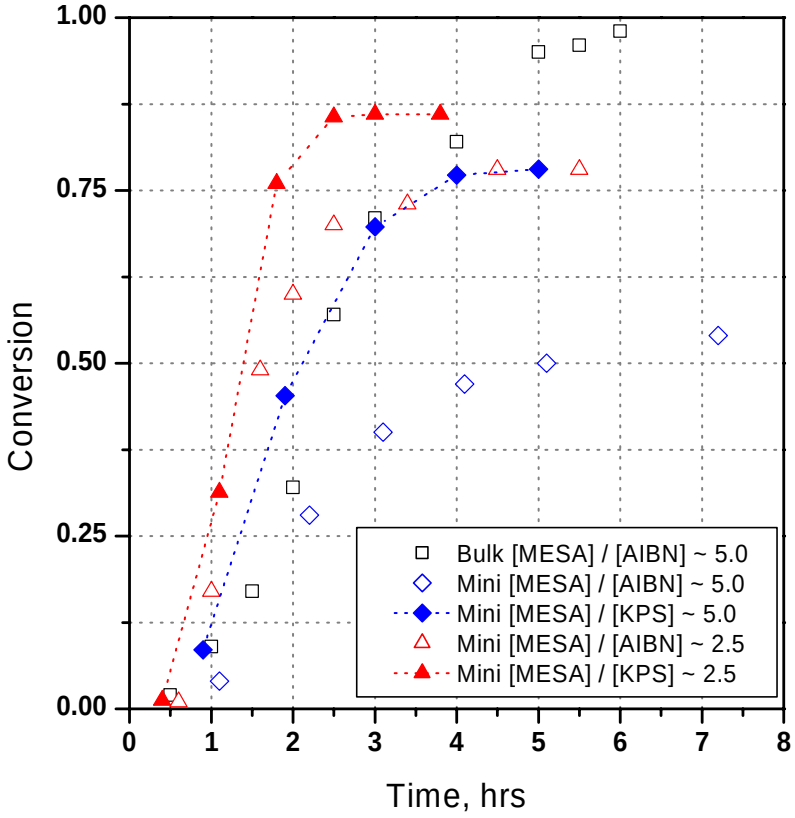


Figure 2.10 Kinetic plot of vinyl acetate RAFT polymerization in miniemulsion comparing the use of an oil-soluble initiator (AIBN) to a water soluble initiator (KPS).

polymerizations failed to achieve similar conversions to the bulk polymerization. The highest conversion was ~ 90% using twice the level of initiator used in the bulk polymerization which achieved almost 100% conversion. As shown in Figure 2.11, the difference in the rate of decomposition of the two initiators is not great enough to account for the higher kinetic rates and overall conversions observed using KPS, particularly at

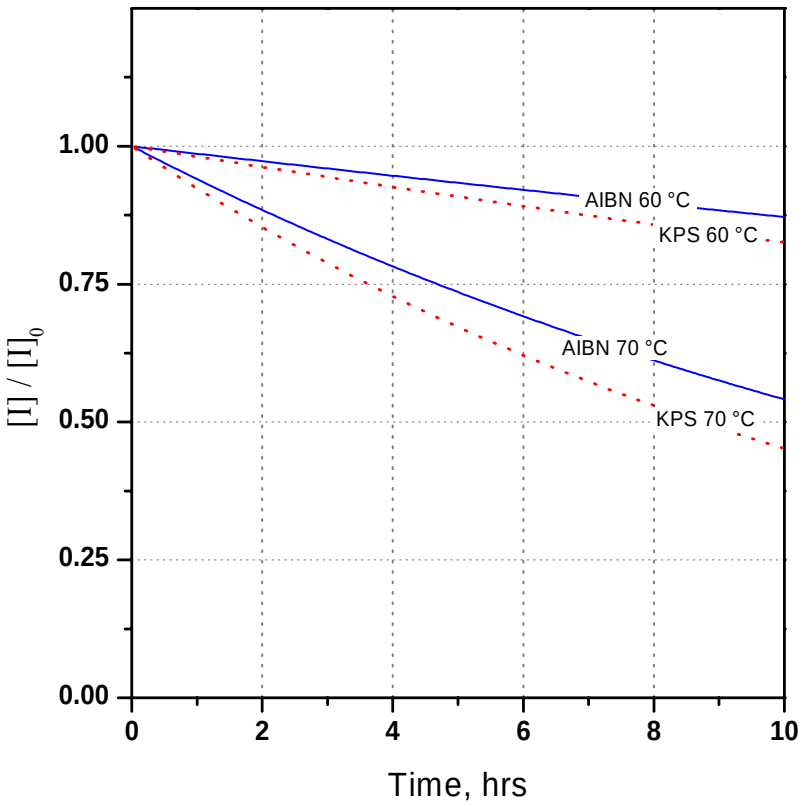


Figure 2.11 Plot comparing the decomposition rates of initiators KPS and AIBN at 60 °C and 70 °C.

60 °C.[63,64] While the results clearly implicate the use of the oil-soluble AIBN initiator in contributing to depressed rates and lower conversion compared to bulk, they also indicate that simply replacing the oil-soluble initiator with a water soluble initiator does not completely address the problem of limiting conversion. The two reports in the literature of homopolymerizations of vinyl acetate in miniemulsion using KPS as initiator[41,42] show conversions of 100%, however initiator levels in each instance were ~5-30 times

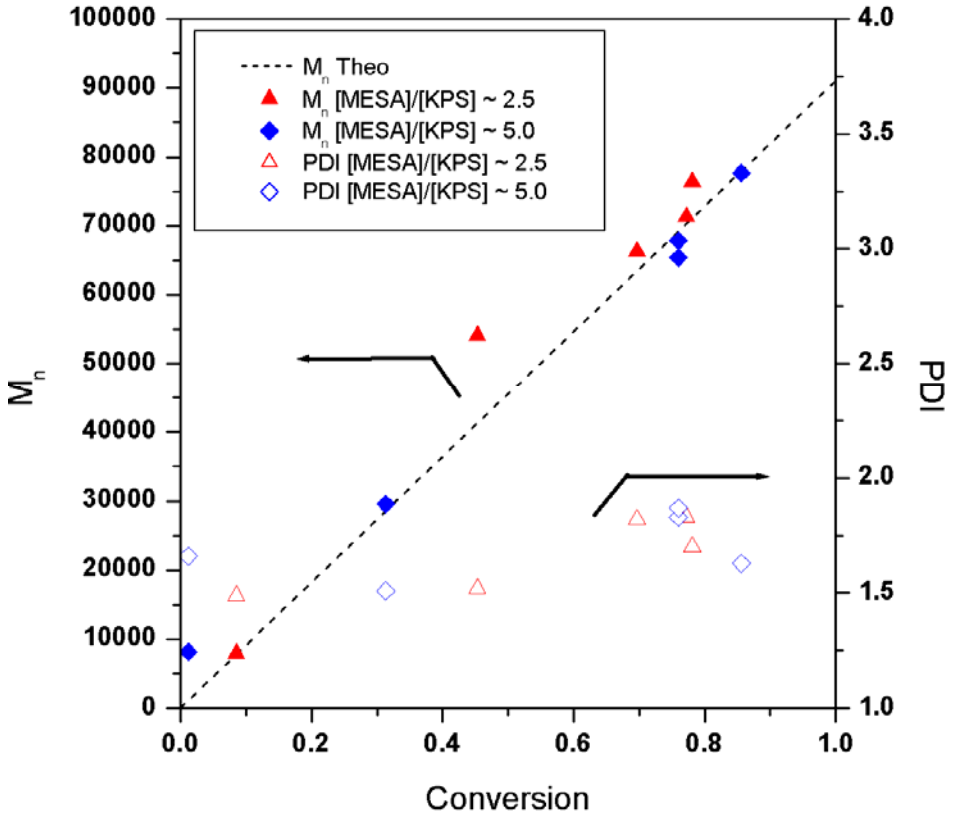


Figure 2.12 Evolution of Mn and PDI as a function of monomer conversion for polymerizations using water soluble initiator, KPS.

the level used here. While increasing the levels of initiator further might overcome the limiting conversion, using concentrations that high with this system would have a negative impact on the control of the polymerization and would certainly broaden the polydispersity.

An examination of the molecular weight evolution (Figure 2.12) shows that the reaction proceeded in a manner consistent with control and the polydispersities were similar to those seen in the polymerizations using AIBN as initiator. The GPC

chromatograms of the polymerization with the highest concentration of initiator ([MESA]/[KPS] ~ 2.5) are shown in Figure 2.13. They indicate that uncontrolled, aqueous phase polymerization appeared to be negligible if present at all, introducing the possibility that water soluble initiators could be employed in these type polymerizations without related detrimental effects. This is provided, of course, that the problem of the limiting conversion can be understood and overcome. Potential courses of action in this regard will be analyzed in the last chapter of this work.

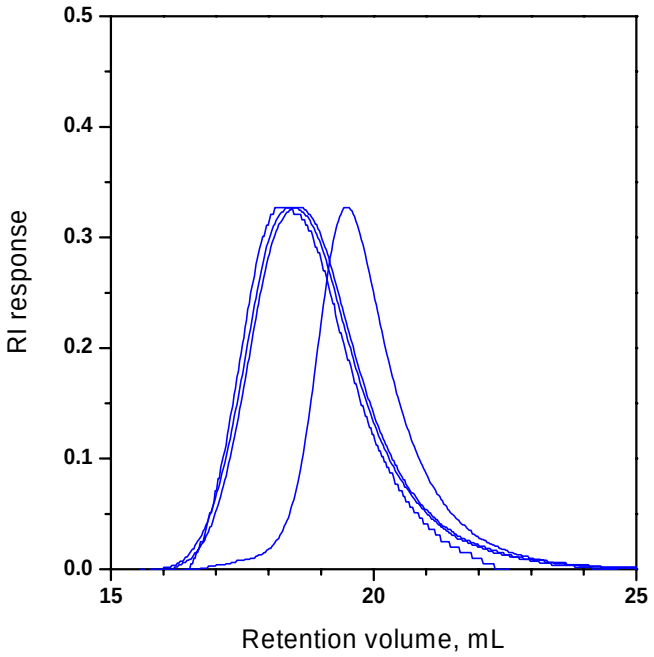


Figure 2.13 GPC traces of polymerization using water soluble initiator, KPS. Chromatograms show no evidence of high molecular weight, uncontrolled polymer

2.3 Conclusions

In this chapter, the controlled polymerization of vinyl acetate in miniemulsion using RAFT chemistry was demonstrated. A linear evolution of number average molecular weight was observed, however the miniemulsion polymerizations exhibited limiting conversions and higher than bulk polydispersities. When an oil-soluble initiator (AIBN) was employed, rates of polymerization were slower in miniemulsion as compared to bulk at the same level of initiator. Additionally, limiting conversions were observed in the AIBN initiated miniemulsion polymerizations. Gel formation, hydrolysis/degradation of the RAFT agent, and desorption of certain molecular species formed early in the reaction were all ruled out as sources and in each case justifying experimental data was presented. The cause was postulated to be related in part to complications from the use of an oil soluble initiator, and experimental data using a water soluble initiator, KPS, was provided giving credence to this hypothesis. GPC traces of the polymer produced indicate that polymer formation in the aqueous phase was negligible whether the initiator used was oil-soluble or water soluble. The use of the water soluble initiator allowed for much improved kinetics and conversions but the polymerizations still exhibited lower conversions than those observed in bulk at the same level of initiator. It was postulated, based on reports in the literature, that higher levels of initiator might serve to overcome the limiting conversions, however the polydispersity of the polymer formed would likely suffer.

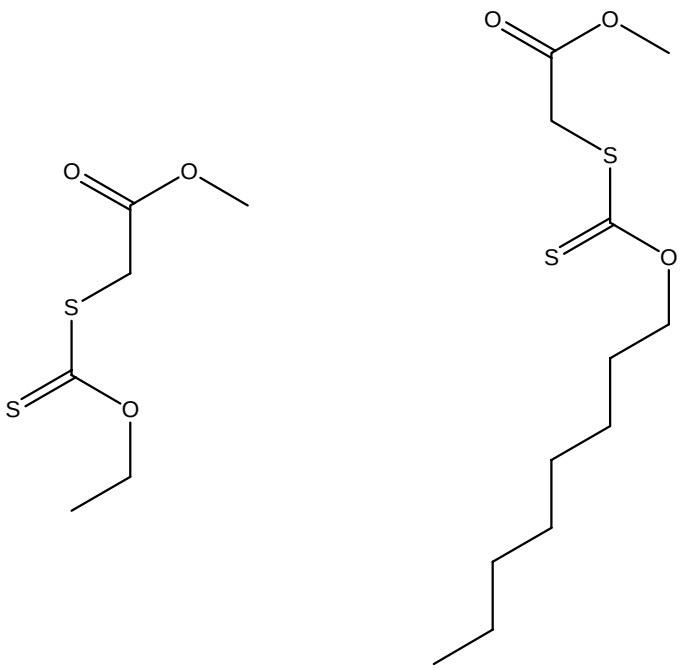
2.4 Experimental Procedures

2.4.1 Materials

Vinyl acetate (VA) was purchased from Aldrich and purified by passing through a column packed with inhibitor remover. AIBN (initiator, 2,2' azo-bis(isobutyronitrile ≥99.0 %, Aldrich), hexadecane (costabilizer, ≥99%, Aldrich), sodium dodecyl sulfate (ionic surfactant, SDS, ≥99%, Aldrich), ethanol (≥99.5%, Aldrich) and octanol (≥99%, Aldrich), carbon disulfide (≥99%, Aldrich) and methyl bromoacetate (≥99%, Aldrich) were used as received. Deionized water was generated in-house with a U.S. Filter Systems Deionizer and was used without further purification.

2.4.2 Synthesis of RAFT agents

Methyl (ethyloxycarbonothioyl)sulfanyl acetate (MESA, see Scheme 2.5) and methyl (octyloxycarbonothioyl)sulfanyl acetate (MOSA, see Scheme 2.5) were prepared using a method modified from Ref [2]. Ethanol (180 g, 3.91 mol) and potassium hydroxide (KOH, 18 g, 0.32 g) were placed in a 250 mL, 3-neck flask under ultra-high purity nitrogen and stirred until the KOH was completely dissolved. Using an addition funnel, carbon disulfide (80 g, 1.05 mol) was slowly added and the mixture stirred for 2 hours. After slowly adding methyl bromoacetate (49 g, 0.32 mol) the mixture was allowed to stir overnight. The precipitate was filter off, and the remaining ethanol was removed via rotovap followed by placing under high vacuum overnight. The product was diluted with diethyl ether and filter over alumina, followed by rotovap and overnight high



MESA MOSA

Scheme 2.5 RAFT agents used in this study

vac to remove the ether. 90% yield on methyl bromoacetate, light yellow liquid. ^{1}H NMR ($CDCL_3$): δ (ppm) 1.38-1.43 (t, 2H, CH_2–C*H*$_3$), 3.74 (s, 3H, CO_2C*H*$_3$), 3.92 (s, 2H, CH_2), 4.6-4.7(q, 3H, C*H*$_2$–CH_3). MOSA was prepared in a similar manner by simply substituting octanol for ethanol.

2.4.3 Bulk polymerization procedure

Vinyl acetate, AIBN (2.1×10^{-3} mol/L, $[MESA]_0/[AIBN]_0 \sim 5$), and MESA (1.1×10^{-2} mol/L, $[VA]_0/[MESA]_0 \sim 1000$) were combined and ~0.5 mL of the solution was

placed in each of ten crimp-top vials. The vials were sealed with septa and deoxygenated by purging with nitrogen for ~5 min. The sealed vials were then placed in an oil bath at 60 °C and were removed at intervals of ~30 minutes. The reactions were quenched by cooling the solutions in an ice bath. Residual monomer was evaporated by placing the vials in a vacuum oven and drying the samples for 24 hours (30 ºC, ~100 kPa vacuum) Monomer conversion was subsequently determined gravimetrically.

2.4.4 Preparation of linear PVA

Vinyl acetate and AIBN (4.0×10^{-5} mol/L) were place in a three-neck flask under nitrogen. The mixture was allowed to de-oxygenate for 30 minutes while agitating with a magnetic stirrer. The flask was then placed in an oil bath at 40 °C and the mixture allowed to polymerized to ~ 4% conversion.[65]

2.4.5 Miniemulsion polymerization procedure

A representative miniemulsion recipe is shown in Table 2.2.The surfactant, SDS was added to water and allowed to mix for 15 minutes. Vinyl acetate, hexadecane, MESA and AIBN were combined and allowed to mix for 15 minutes. The organic phase was then added to the aqueous phase and agitated vigorously with a magnetic stirrer for 10 minutes, forming a very faint yellow emulsion. The miniemulsion was formed by sonicating for 20 minutes (Fisher 300 Sonic Dismembrator at 70% output). During the sonication, the miniemulsion was cooled by an ice bath in order to suppress any thermal initiation. After sonication, the miniemulsion was transferred to a 250 mL, round-bottomed, 3-neck flask outfitted with a septum, reflux condenser, nitrogen feed and

thermometer to monitor the temperature of the miniemulsion. The miniemulsion was kept agitated by a magnetic stirrer. After allowing the miniemulsion to de-oxygenate under ultra high purity nitrogen for 30 minutes, the flask was immersed in an oil bath that

Table 2.2 Representative recipe for the batch miniemulsion polymerization of vinyl acetate.

Component		Mass	Basis
Water		80.0 g	
Monomer	Vinyl acetate	20.0 g	25 wt-% of water
Surfactant	SDS	0.45 g	0.018 mol/L (based on aqueous phase)
Costabilizer	Hexadecane	0.40 g	2 wt-% of monomer
RAFT agent	MESA	0.045 g	$[VA]_0/[MESA]_0 = 1000$
Initiator	AIBN	0.008 g	$[MESA]_0/[AIBN]_0 = 5$

had been preheated to the desired reaction temperature. Samples were withdrawn through the septum via syringe at regular intervals for conversion, GPC, and particle size analysis. The reaction was kept under nitrogen for the entire time of the experiment.

2.4.6 MESA decomposition/hydrolysis experiments

Table 2.3 shows the recipes used for the three hydrolysis experiments. The basis was the same as for the miniemulsion experiments. The reagents were first placed in a 50 mL, 3-neck flask equipped with a septum and condenser. They were agitated with a magnetic stirrer and kept under ultra-pure nitrogen. After purging for 30 minutes, the flask was placed in an oil bath at 60 °C. Samples were withdrawn through the septum via syringe at regular intervals for analysis. After drying for 24 hours in a vacuum oven (30 ºC, ~ 100 kPa vacuum), they were dissolved in THF and analyzed using SEC/UV.

Table 2.3 Recipe for hydrolysis experiments

Compound	Exp		
	DCMP 1	DCMP 2	DCMP 3
Water	25 g	25 g	25 g
MESA	0.014 g	0.014 g	0.014 g
VA		6.25 g	
SDS	0.14 g	0.14 g	0.14 g
Hexadecane		0.125 g	
AIBN			0.047 g

2.4.7 Seeded miniemulsion using MOSA

The recipe is shown in Table 2.4. The entire procedure was conducted using a Schlenk line and under argon in order to prevent oxygen contamination. The organics were first placed in a three-neck 50 mL flask under argon with magnetic stirring, purged for 30 minutes, and placed in an oil bath at 60 °C. After polymerizing to ~ 8% conversion, the organics were transferred under argon using Schlenk techniques to a sealed sonication vessel which contained the water and surfactant. After sonicating for 20 minutes to form the miniemulsion, the mixture was transferred under argon to a clean flask, heated to 60 °C using an oil bath allowing the polymerization to continue.

Table 2.4 Recipe for the seeded miniemulsion polymerization of VA using MOSA

Component		Mass	Basis
Water		80.0 g	
Monomer	Vinyl acetate	20.0 g	25 wt-% of water
Surfactant	SDS	0.45 g	0.018 mol/L (based on aqueous phase)
Costabilizer	Hexadecane	0.40 g	2 wt-% of monomer
RAFT agent	MOSA	0.068 g	$[VA]_0/[MOSA]_0 = 1000$
Initiator	AIBN	0.008 g	$[MOSA]_0/[AIBN]_0 = 5$

2.4.8 Characterization

Polymer samples were dried for 24 hours in a vacuum oven (30 ºC, ~ 100 kPa vacuum) and monomer conversion was subsequently determined gravimetrically. The number average molecular weight, M_n, and the polydispersity, M_w/M_n, were calculated using data gathered via size exclusion chromatography (SEC-Viscometry-RALLS)[66] with THF as eluent. Three columns (American Polymer Standards styrene-divinylbenzene 100 Å, 1000 Å, and 10^5 Å) mounted in a Waters WAT038040 column heater set at 30 ºC were utilized. The columns were connected to a Viscotek GPCMax pump/autoinjector, a Viscotek T60A dual detector (viscometer and light scattering), a Waters 410 refractive index detector, an LDC Milton Roy Spectromonitor 3000 UV detector (at 311 nm). Latex particle sizes and polydispersities were analyzed using quasi-elastic light scattering (QELS, Protein Solutions DynaPro99 with DynaPro DCS v 5.26 software).

2.5 References

1. Rizzardo, E.; Chiefari, J.; Mayadunne, R. T. A.; Moad, G.; Thang, S. H. *ACS Symp. Ser.* **2000**, *768*, 278.

2. Stenzel, M. H.; Cummins, L.; Roberts, G. E.; Davis, T. R.; Vana, P.; Barner-Kowollik, C. *Macromol. Chem. Physic.* **2003**, *204*, 1160.

3. Destarac, M.; Charmot, D.; Franck, X.; Zard, S. Z. *Macromol. Rapid Comm.* **2000**, *21*, 1035.

4. Coote, M. L.; Radom, L. *Macromolecules* **2004**, *37*, 590.

5. Favier, A.; Barner-Kowollik, C.; Davis, T. P.; Stenzel, M. H. *Macromol. Chem. Physic.* **2004**, *205*, 925.

6. Stenzel, M. H.; Davis, T. P.; Barner-Kowollik, C. *Chem. Commun.* **2004**, 1546.

7. Destarac, M.; Taton, D.; Zard, S. Z.; Saleh, T.; Yvan, S. *ACS Symp. Ser.* **2003**, *854*, 536.

8. Charmot, D.; Corpart, P.; Adam, H.; Zard, S. Z.; Biadatti, T.; Bouhadir, G. *Macromol. Symp.* **2000**, *150*, 23.

9. Simal, F.; Delfosse, S.; Demonceau, A.; Noels, A. F.; Denk, K.; Kohl, F. I.; Weskamp, T.; Herrmann, W. A. *Chem.-Eur. J.* **2002**, *8*, 3047.

10. Wakioka, M.; Baek, K. Y.; Ando, T.; Kamigaito, M.; Sawamoto, M. *Macromolecules* **2002**, *35*, 330.

11. Xia, J. H.; Paik, H. J.; Matyjaszewski, K. *Macromolecules* **1999**, *32*, 8310.

12. Russum, J. P.; Barbre, N. D.; Jones, C. W.; Schork, F. J. *J. Polym. Sci. Polym. Chem.* **2005**, *43*, 2188.

13. El-Aasser, M. S.; Vanderhoff, J. W., *Emulsion Polymerization of Vinyl Acetate*. Applied Science Pub.: London, 1981.

14. Simms, R. W.; Davis, T. P.; Cunningham, M. F. *Macromol. Rapid Comm.* **2005**, *26*, 592.

15. Butté, A.; Storti, G.; Morbidelli, M. *Macromolecules* **2001**, *34*, 5885.

16. Gilbert, R. G., *Emulsion Polymerization : A Mechanistic Approach*. Academic Press: 1995.

17. Tsavalas, J. G.; Luo, Y. W.; Hudda, L.; Schork, F. J. *Polym. React. Eng.* **2003**, *11*, 277.

18. Brandrup, J.; Immergut, E. H., *Polymer Handbook*. 3rd ed.; Wiley: 1989.

19. Wheeler, O. L.; Ernst, S. L.; Crozier, R. N. *J. Polym. Sci.* **1952**, *8*, 409.

20. Wheeler, O. L.; Lavin, E.; Crozier, R. N. *J. Polym. Sci.* **1952**, *9*, 157.

21. Zimm, B. H.; Stockmayer, W. H. *J. Chem. Phys.* **1949**, *17*, 1301.

22. Grcev, S.; Schoenmakers, P.; Iedema, P. *Polymer* **2004**, *45*, 39.

23. Podzimek, S. *J. Appl. Polym. Sci.* **1994**, *54*, 91.

24. Wyatt, P. J. *Anal. Chim. Acta* **1993**, *272*, 1.

25. Thomas, D. B.; Sumerlin, B. S.; Lowe, A. B.; McCormick, C. L. *Macromolecules* **2003**, *36*, 1436.

26. Thomas, D. B.; Convertine, A. J.; Hester, R. D.; Lowe, A. B.; McCormick, C. L. *Macromolecules* **2004**, *37*, 1735.

27. Levesque, G.; Arsene, P.; Fanneau-Bellenger, V.; Pham, T. N. *Biomacromolecules* **2000**, *1*, 400.

28. Baussard, J. F.; Habib-Jiwan, J. L.; Laschewsky, A.; Mertoglu, M.; Storsberg, J. *Polymer* **2004**, *45*, 3615.

29. Mertoglu, M.; Laschewsky, A.; Skrabania, K.; Wieland, C. *Macromolecules* **2005**, *38*, 3601.

30. Krstina, J.; Moad, G.; Willing, R. I.; Danek, S. K.; Kelly, D. P.; Jones, S. L.; Solomon, D. H. *Eur. Polym. J.* **1993**, *29*, 379.

31. Barbe, W.; Ruchardt, C. *Makromol. Chem.* **1983**, *184*, 1235.

32. Fink, J. K. *J. Polym. Sci. Pol. Chem.* **1983**, *21*, 1445.

33. Jaffe, A. B.; Skinner, K. J.; Mcbride, J. M. *J. Am. Chem. Soc.* **1972**, *94*, 8510.

34. Krstina, J.; Moad, G.; Solomon, D. H. *Eur. Polym. J.* **1989**, *25*, 767.

35. Moad, G.; Rizzardo, E.; Solomon, D. H.; Johns, S. R.; Willing, R. I. *Makromol. Chem.-Rapid* **1984**, *5*, 793.

36. Moad, G.; Solomon, D. H.; Johns, S. R.; Willing, R. I. *Macromolecules* **1984**, *17*, 1094.

37. Serelis, A. K.; Solomon, D. H. *Polymer Bulletin* **1982**, *7*, 39.

38. Wang, S.; Schork, F. J. *J. Appl. Polym. Sci.* **1994**, *54*, 2157.

39. Aizpurua, I.; Barandiaran, M. J. *Polymer* **1999**, *40*, 4105.

40. Aizpurua, I.; Amalvy, J. I.; de la Cal, J. C.; Barandiaran, M. J. *Polymer* **2000**, *42*, 1417.

41. Aizpurua, I.; Amalvy, J. I.; Barandiaran, M. J. *Colloid. Surface. A* **2000**, *166*, 59.

42. Wu, X. Q.; Schork, F. J. *J. Appl. Polym. Sci.* **2001**, *81*, 1691.

43. Graillat, C.; Guyot, A. *Macromolecules* **2003**, *36*, 6371.

44. Silva, F. M.; Lima, E. L.; Pinto, J. C. *Ind. Eng. Chem. Res.* **2004**, *43*, 7324.

45. Zhang, S. X.; Ray, W. H. *Ind. Eng. Chem. Res.* **1997**, *36*, 1310.

46. Luo, Y. W.; Tsavalas, J.; Schork, F. J. *Macromolecules* **2001**, *34*, 5501.

47. Choi, Y. T.; El-Aasser, M. S.; Sudol, E. D.; Vanderhoff, J. W. *J. Polym. Sci. Polym. Chem. Ed.* **1985**, *23*, 2973.

48. Asua, J. M.; Sudol, E. D.; El-aasser, M. S. *J. Polym. Sci. Pol. Chem.* **1989**, *27*, 3903.

49. Alduncin, J. A.; Forcada, J.; Asua, J. M. *Macromolecules* **1994**, *27*, 2256.

50. Asua, J. M.; Rodriguez, V. S.; Sudol, E. D.; El-aasser, M. S. *J. Polym. Sci. Pol. Chem.* **1989**, *27*, 3569.

51. Bartoň, J.; Karpatyova, A. *Makromol. Chem.* **1987**, *188*, 693.

52. Nomura, M.; Fujita, K. *Makromol. Chem.-Rapid* **1989**, *10*, 581.

53. Nomura, M.; Ikoma, J.; Fujita, K. *J. Polym. Sci. Pol. Chem.* **1993**, *31*, 2103.

54. Nomura, M.; Suzuki, K. *Ind. Eng. Chem. Res.* **2005**, *44*, 2561.

55. Alduncin, J. A.; Asua, J. M. *Polymer* **1994**, *35*, 3758.

56. Reimers, J. L.; Schork, F. J. *Ind. Eng. Chem. Res.* **1997**, *36*, 1085.

57. Chern, C. S.; Liou, Y. C. *J. Polym. Sci. Pol. Chem.* **1999**, *37*, 2537.

58. Blythe, P. J.; Klein, A.; Phillips, J. A.; Sudol, E. D.; El-Aasser, M. S. *J. Polym. Sci. Pol. Chem.* **1999**, *37*, 4449.

59. Ghazaly, H. M.; Daniels, E. S.; Dimonie, V. L.; Klein, A.; El-Aasser, M. S. *J. Appl. Polym. Sci.* **2001**, *81*, 1721.

60. Luo, Y. W.; Schork, F. J. *J. Polym. Sci. Pol. Chem.* **2002**, *40*, 3200.

61. Kim, N.; Sudol, E. D.; Dimonie, V. L.; El-Aasser, M. S. *Macromolecules* **2004**, *37*, 3180.

62. Patrick, C. R.; Allen, P. E. M. *Eur. Polym. J.* **1965**, *1*, 247.

63. Ham, G. E., *Kinetics and Mechanisms of Polymerization*. M. Dekker: New York, NY, 1967; Vol. 1.

64. Lansalot, M.; Davis, T. P.; Heuts, J. P. A. *Macromolecules* **2002**, *35*, 7582.

65. Uy, W. C.; Graessley, W. W. *Macromolecules* **1971**, *4*, 458.

66. The SEC-Viscometry_RALLS (Right Angle Laser Light Scattering) technique allows for the absolute characterization of polymers. In principle, it requires only that the concentration of the sample be known.

CHAPTER 3

CONTINUOUS RAFT POLYMERIZATION IN MINIEMULSION UTILIZING A MULTI-TUBE REACTION SYSTEM[†]

3.1 Introduction

As pointed out in the first chapter, demonstrating that RAFT could be combined with a continuous, easily variable process that is environmentally sensible in nature would do much to increase its commercial viability. Since miniemulsions are currently the most promising aqueous dispersed systems for conducting CRPs, combining RAFT with continuous miniemulsion is a reasonable progression towards that end. In an effort to demonstrate the feasibility of such a system, herein is described a reactor system that facilitates rapid data collection in the study of continuous RAFT/miniemulsion.[1,2] Because one of the hallmarks of a CRP is the linear evolution of the number average molecular weight with conversion, the ability to collect data throughout the entire reaction in a practical manner is desirable. The conservation of RAFT agent is also a concern, making small reaction volumes a priority. To overcome these two challenges, an atypical and unique tubular reactor system for the study of continuous RAFT/miniemulsion polymerization was designed, built and implemented. The system employs multiple tubes of different lengths in order to produce residence times at various points along the span of the reaction curve. Small diameter tubing is used so as to keep the total reactor volume to a minimum, therefore conserving RAFT agent. Also presented

[†] Portions of this chapter have been previously published, *Ind. Eng. Chem. Res.* **2005**, *44*, 2484 and *Macromol. Rapid Commun.* **2004**, *25*, 1064.

is the development of a recipe using mixed surfactants to impart stability and eliminate phase separation of the miniemulsion, the results of styrene homopolymerizations in batch and tube, and the results of a chain extension experiment demonstrating the living nature of the chains by forming a copolymer.

In conducting these studies, two fundamental questions were the focus. Kinetically, would the tube reactor behave in a similar fashion as a batch reactor, provided identical reaction parameters and plug or near plug flow? In theory, in the limit of plug flow, they should be the same, with the time variable replaced by the length of the tube.[3] Secondly, what is the effect of laminar flow or of significant axial dispersion on the final polydispersity? In laminar flow or when the axial dispersion is high, there will be a distribution of particle residence times[3] which in principle would contribute to an increase in the final polydispersity of the polymer.

3.2 Results and Discussion

3.2.1 Initial batch polymerizations with PEPDTA

Before performing homopolymerizations in the tube reactor, it was necessary to develop and test a suitable recipe for the miniemulsion. A typical recipe is shown in Table 3.1. The focus was on the choice of an appropriate monomer/RAFT agent/surfactant combination. Initially, PEPDTA was selected because when used in conjunction with styrene it has a sufficiently high transfer constant (est. ~ 130)[4] to mediate a well-controlled reaction but likely low enough to avoid problems with latex instability.[5] It has been used in miniemulsions and the researchers reported no stability

Table 3.1. Recipe for the batch miniemulsion polymerization of styrene.

Component	Mass, g	Basis
Water	50.0 g	
Styrene	12.5 g	25 wt-% of water
Triton X-405	0.58 g	0.005 mol/L (based on aqueous phase)
SDS	0.07 g	"
Hexadecane	0.25 g	2 wt-% of monomer
PEPDTA	0.11 g	$[Sty]_0/[PEPDTA]_0 = 300$
KPS	0.01 g	$[PEPDTA]_0/[KPS]_0 = 10$
Temperature	70 ºC	

issues when combined with SDS as surfactant.[4] However, minor phase separation visible as a yellow organic layer on top of the miniemulsion (~ 5-10% of the total monomer, by visual inspection) was observed in approximately 50% of preliminary test experiments. Based partially on prior successes combining RAFT with nonionic surfactants,[6] a dual surfactant system using sodium dodecyl sulfate (SDS, an ionic surfactant) and Triton® X-405 (TX405, a non-ionic, polymeric surfactant), was utilized in order to overcome these stability issues. Some versatility was also desired, for example if a different monomer were to be employed with PEPDTA that may result in a higher transfer constant which in turn could affect the stability.[7] Colombié et al.[8] showed that in certain concentrations, strong adsorption of both surfactants on polystyrene particles leads to larger total coverage than would be obtained with either surfactant acting alone. Greater coverage in principle should provide increased thermodynamic stability to the droplets in the critical first stages of the polymerization.

As shown in Figure 3.1, the number average molecular weight increased linearly with conversion, with very little deviation from the theoretical prediction. The fact that the molecular weights at very low conversions were close to theoretical suggests that the transfer constant of PEPDTA with styrene is indeed high enough to insure that the RAFT agent is incorporated into the growing chains sufficiently early in the reaction. The final polydispersities were also relatively low, ~ 1.35, and the progression of the polydispersity

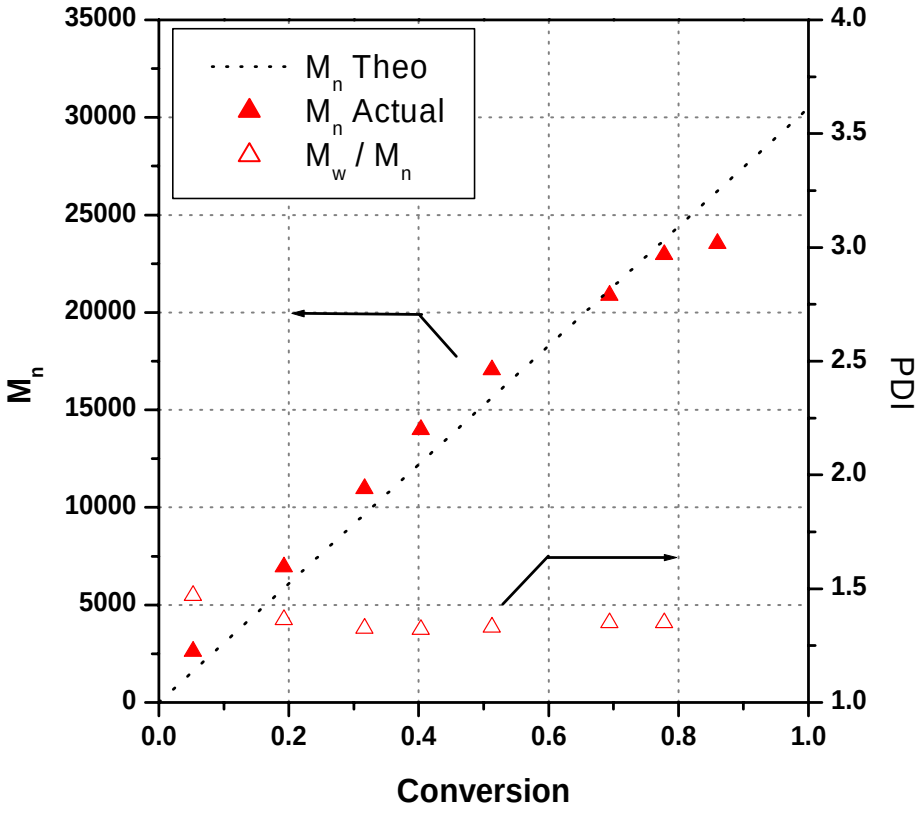


Figure 3.1. Plot of the number average molecular weight and polydispersity as a function of conversion of polystyrene produced via the batch RAFT/miniemulsion.

from higher values early in the reaction to lower values is consistent with a controlled polymerization. An analysis of the GPC chromatograms (see Figure 3.2) shows a shift towards increasing molecular weights in the distributions, characteristic of CRPs. In each chromatogram a mono-modal distribution was observed indicating little or no free radical polymerization in the aqueous phase. The reaction rate began to decrease at ~ 70-80% conversion, however the first order plot of monomer conversion shown in Figure 3.3 reveals that bi-molecular termination does not play a significant role in the rate reduction and it is more likely owing simply to the depletion of monomer in the particles. It should

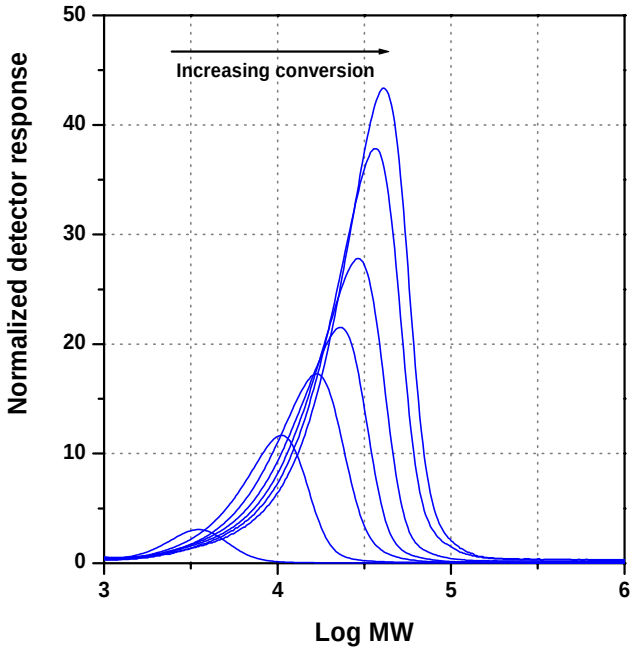


Figure 3.2 Experimental MWD curves at increasing conversions for polystyrene batch RAFT/miniemulsion

be noted that with respect to RAFT, a first order plot of monomer conversion cannot be used independently as an indicator of control, since the radical concentration can change in a RAFT polymerization.[9,10] It can, however, be employed diagnostically as a gauge

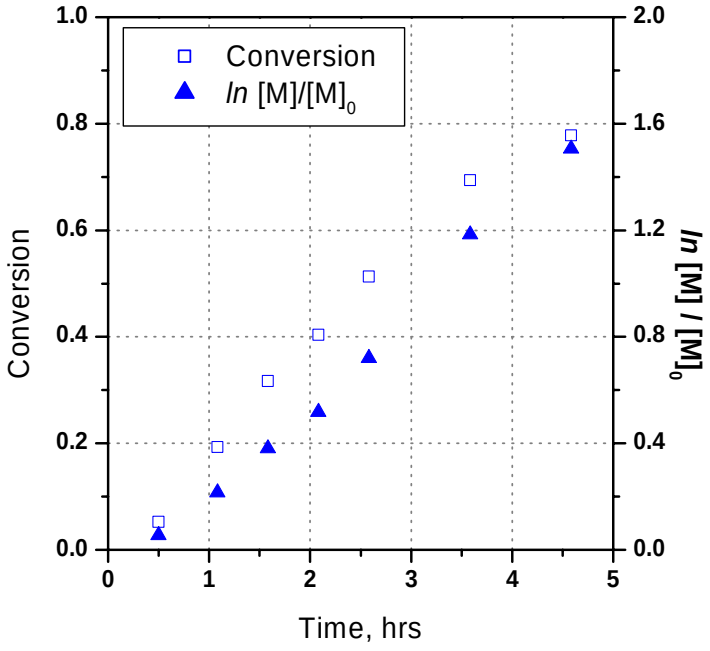


Figure 3.3 Combined conversion vs. time and first order plot of polystyrene batch RAFT/miniemulsion

of the relative extent of the initiation and termination reactions and that is the manner in which it used here. Given the fact that the half-life of the initiator, KPS, at 70 ºC is approximately 8 hours it is unlikely that the radical flux has changed significantly, meaning any decrease in the overall rate as a result of increased termination would likely be reflected in the $\ln[M]/[M]_0$ plot. What is interesting to note is that the kinetic data point to an apparent inhibition period of approximately 15-20 minutes. The cause is likely

not inhibition, but rather the selectivity of the RAFT agent the during the initial stages of the reaction.[11,12] This selectivity, which is attributable a large difference between the addition and fragmentation rate coefficients during the pre-equilibrium stage of the reaction, heavily favors the addition of R-group and initiator derived radicals. As such, the formation of monomer-derived oligomers is suppressed until most of the RAFT agent is consumed. An important and encouraging observation was the excellent stability of the miniemulsion. No separation or coagulation was observed at any time during the course of the experiment. Additionally, there was no evidence of secondary, or aqueous, particle nucleation, as evidenced by the ratio of the final and initial number of particles, which was calculated to be ~ 0.96 (based on the initial and final particle sizes). This gives further support to the observations made earlier about the molecular weight distributions. As Table 3.1 shows, the initial surfactant concentration/CMC ratio of the TX405 was 6.2 whereas with SDS the ratio was 0.6. This suggests that in the extreme if none of the SDS

Table 3.2 Surfactant concentrations, critical micelle concentrations and concentration ratios

Surfactant	Concentration $mol \cdot L^{-1}$	CMC $mol \cdot L^{-1}$	$[Surf]_0$/CMC Ratio
SDS	0.005	0.009	0.6
Triton X-405	0.005	0.00081	6.2

adsorbed to the droplets and all of the TX405, no micelles should be formed. In contrast, the nonionic surfactant TX405 has a very low critical micelle concentration, ~ 8.1×10^{-4}

mol·L^{-1},[13] and because it was present in relatively large amount if only a small percentage (~ 16%) were not adsorbed on the surface of the droplets, those remaining molecules could in principle form micelles in the aqueous phase and become a source for secondary nucleation. If significant, this would likely reveal itself in the GPC chromatograms as bimodal distribution of both low molecular weight controlled polymer and high molecular weight, uncontrolled polymer. As already mentioned, this was not observed. While the absence of the high polymer does not preclude the existence of micelles, it does suggest that any micellar formation was negligible and that most of the surfactant resided on the surface of the droplets.

3.2.2 Continuous polymerizations

Recipe and reaction parameters for a typical styrene homopolymerization in the tubular reactor are shown in Table 3.3 and Table 3.4, respectively. Flow rates were set to ~0.2-0.35 mL/min depending on the desired residence time. Visual inspection of the clear tubing afterward revealed no fouling of the tubes. In order to insure the reactor was operating at steady state, samples were taken over several residence times in each tube. This is demonstrated for experiment T3 in Figure 3.4 which shows the conversion plotted against the dimensionless residence time. While some transient behavior is noted in tubes 2, 3, and 4, all of the tubes reached steady state after ~3 residence times. All of the samples shown here for kinetic and molecular weight analysis for the tubular reactions were taken after at least three residence times. An analysis of the kinetic data shown in Figure 3.5 reveals that similar to the original batch trials of the recipe, a period of apparent inhibition is noticeable in both the tube and batch experiments. As discussed in

the previous section, this is likely owing to the selectivity of the RAFT radicals during the initial, pre-equilibrium, phase of the reaction.[11,12]

However, in order to scrutinize whether residual inhibitor in the monomer might instead be the cause, experiments T1 and T2 were performed with distilled monomer and with monomer that was cleaned with a column packed with inhibitor remover, respectively. There were no noticeable effects on the apparent inhibition in the polymerizations. If the cause were residual inhibitor or some impurity in the monomer, it is highly probable that inhibition would have been eliminated, or at least significantly reduced, in one of the two experiments.

Table 3.3 Recipe for the miniemulsion polymerization of styrene in tubular reactor

Component	Mass	Basis
Water	1600 g	
Styrene	400 g	25 wt-% of water
Triton X-405	18.5 g	0.005 mol/L (based on aqueous phase)
SDS	2.3 g	"
Hexadecane	8.0 g	2 wt-% of monomer
PEPDTA	3.5 g	$[Sty]_0/[PEPDTA]_0 = 300$
KPS	0.35 g	$[PEPDTA]_0/[KPS]_0 = 10$
Temperature	70 ºC	

Table 3.4 Summary of reactor parameters

Tube	Length (m)	Flow (mL/min)	Residence time (min)	Reynolds number
1	8.1	0.29	55	9.3
2	15.5	0.30	105	9.4
3	23.2	0.30	153	9.5
4	31.0	0.34	180	10.8
5	38.9	0.34	226	10.8

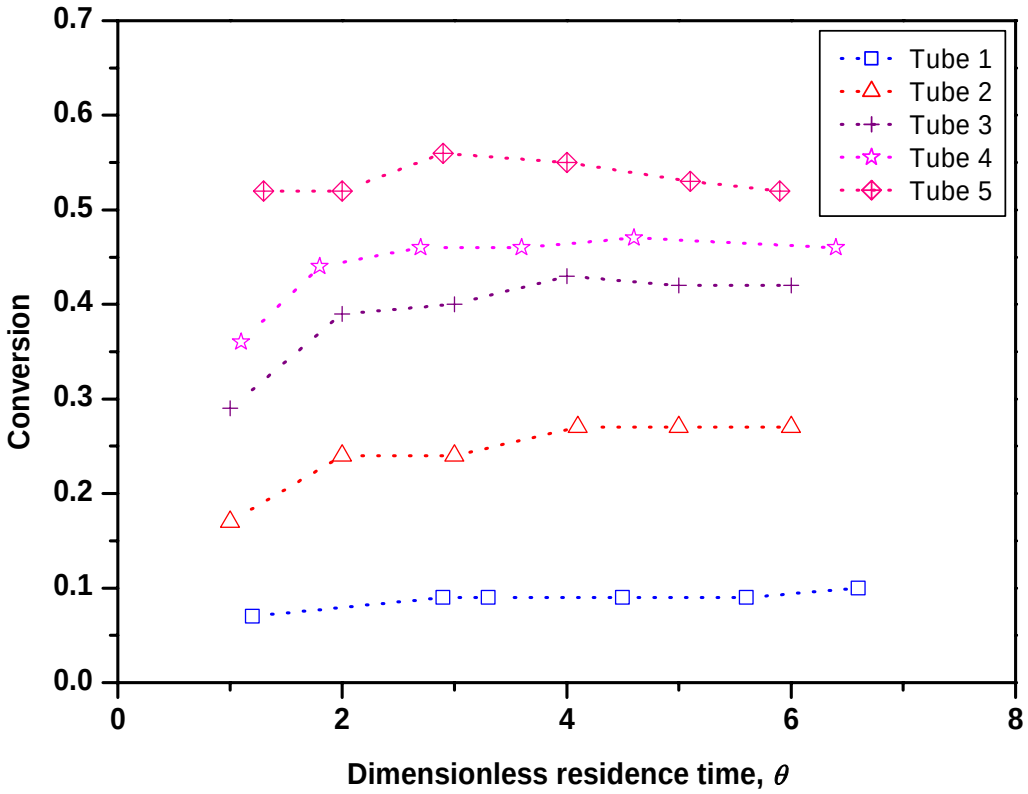


Figure 3.4 Plot of the conversion (Exp T3) as a function of the dimensionless residence time in each of the tubes in the reactor

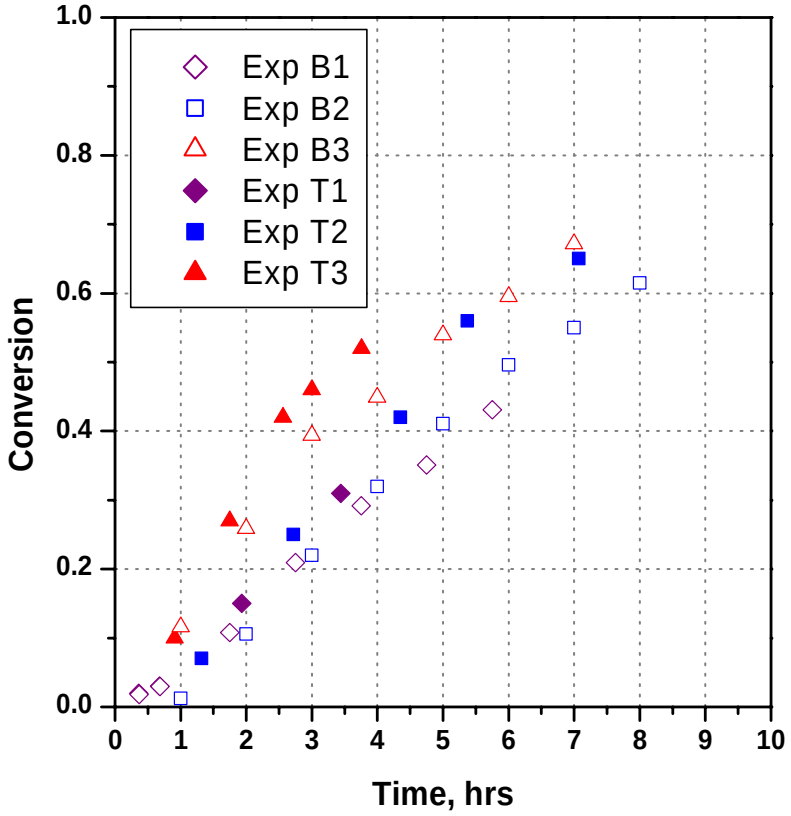


Figure 3.5 Plot of the kinetic profile of the reactions in the tubular reactor along with their corresponding concurrent batch reaction.

Given the flow regime in the tubular reactor, with Re ~ ≤ 10, the effects of a residence time distribution, either from a laminar flow velocity gradient or axial dispersion, might be expected to have an effect on the kinetics. If this were the case, then in the absence of other effects a lower reaction rate should be observed in the tube as compared to batch. The degree to which this would occur would be dictated by the amount of dispersion and in the high limit should approach mixed flow. The most interesting observation about the kinetics, however, is that while very similar, in each

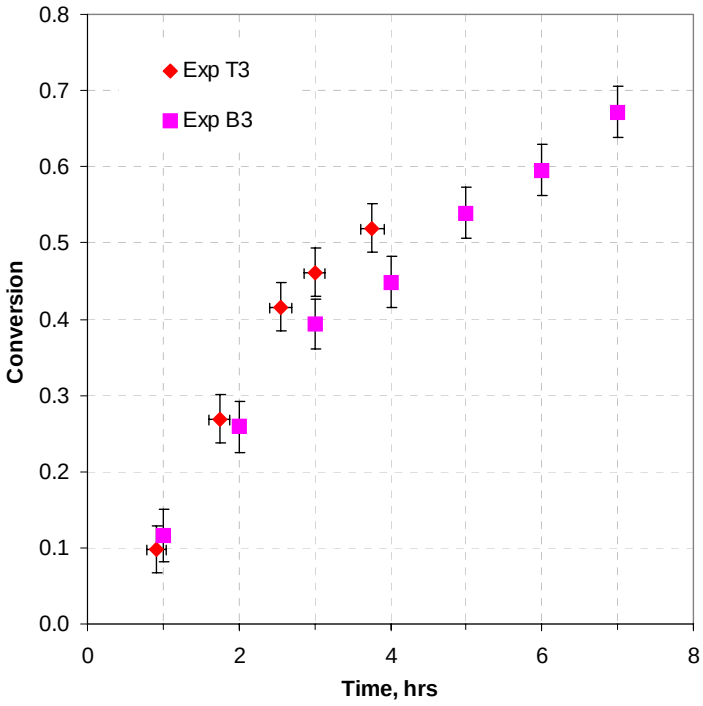


Figure 3.6 Kinetic plot of concurrent tube and batch experiment with error bars. The *x*-error bars associated with the tube experiment (Exp T3) represent the measurement error in the flow rates.

case the tube experiments were slightly faster than those in batch (see Figure 3.5). As mentioned earlier, flow remained consistent after 2 or 3 residence times as indicated by the steady state profile of the tubular reactor as well as measured flow rates. Propagation of the flow measurement error, shown in Figure 3.6 for T3, indicates that something else is most likely at play. Initially it was thought that slight temperature differences between the batch and tube experiments might also be at the root of the difference. Temperatures were maintained with controllers, however they were not independently monitored for temperature (i.e., checking the temperature of each system with the same thermometer or temperature probe), allowing for the possibility that a small difference in temperature might exist between the two systems. At 70 ºC, a ±1 ºC temperature difference can result

in a ± 5% variation in initiator concentration after 4 hours. However at the temperatures and particle sizes seen here, that only translates to ± 3.0-3.5% difference in the value of $\bar{n}$, the average number of radicals per particle. This is not sufficient to explain the kinetic difference seen here.

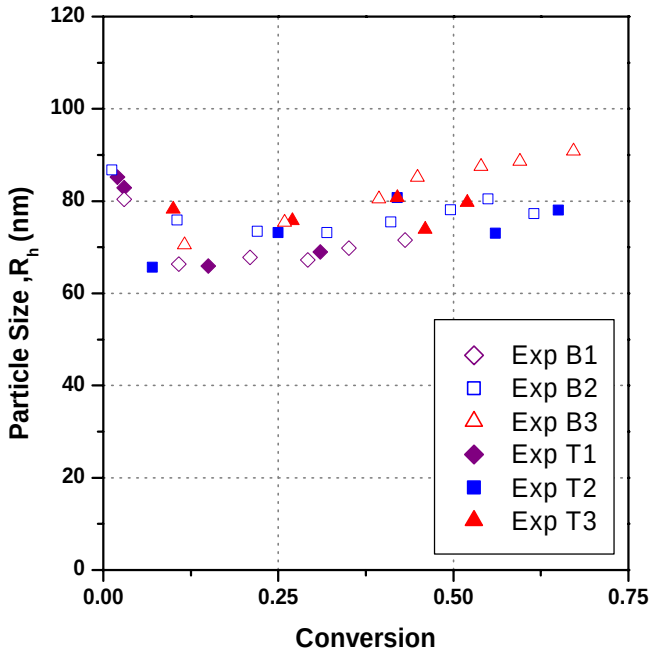


Figure 3.7 Particle size evolution as a function of conversion in tube and batch latex of polystyrene produced via the RAFT/miniemulsion.

With miniemulsions, differences in particle size can cause variations in polymerization rates,[14] but since the batch and tube miniemulsions were essentially the same in preparation, significant particle size differences would not be expected. Figure 3.7 shows the particle radius evolution in each of the experiments. The slight upward trend in the size evolution is most likely owing to the necessity to dilute the latex (~22000:1) in order to perform the light scattering measurements. Monomer left in the

particles would tend to partition to the aqueous phase and thus artificially shrink the size of the particle, which would be especially pronounced at low conversions. In any event, this further complicates using the particle size assessment as a quantitative kinetic parameter. Because of this uncertainty and the fact that the data contain a bit of scatter it is difficult to pinpoint whether or not a significant enough particle size difference exists between the batch and tube experiments. What can be stated is that at the reaction conditions here, changes in the average particle radius of as little ± 5nm can change $\bar{n}$ by ±15-20%, which would be reflected in higher (or lower) rates of polymerization.[14] As such, kinetic variations owing to differences in particle size between batch and tube can not be entirely ruled out as a possibility.

It was noted during the tube experiments that the feed leaving the emulsion feed flask (see Figure 3.15 in Section 3.4.4) separated somewhat before entering the sonication vessel. Because of the geometry of the tubing, it was thought this separation caused a larger fraction of the organic phase to be pumped into the sonication vessel, disturbing the initial mass fractions of the mixture. In order to probe whether or not this may be the source of the kinetic difference between batch and tube, the tube reactor was re-configured, eliminating the emulsion feed tank and feeding the organic and aqueous phase by pump separately and directly into the sonication vessel. In this manner, a consistent miniemulsion could be produced and fluctuations in the individual component mass fractions eliminated. The results of this experiment, T4/B4 are shown in Figure 3.8. In this case, the batch and tube kinetics are almost identical, strong evidence that the source of the difference in the first three experiments was equipment related. If the mass basis of the organic phase were to increase because of the phase separation that was

noted, then calculating the conversion based on a lower, initial mass fraction would artificially inflate the conversion. By changing the way that the miniemulsion was formed and insuring that the mass fractions remained consistent, the variation was eliminated.

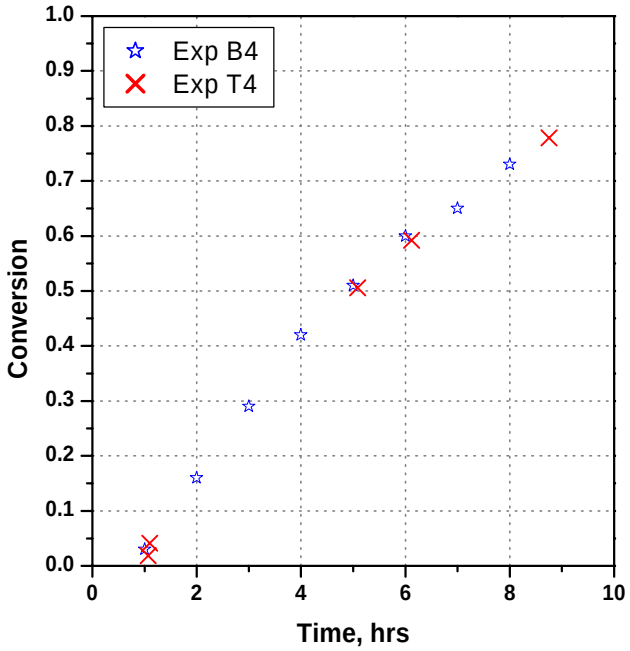


Figure 3.8 Plot comparing batch and tube kinetics in re-configured tubular reactor.

As for evidence of control, the molecular weight results of three different experiments in the tubular reactor are shown in Figure 3.9. In each case the number average molecular weight progressed linearly with conversion, with little deviation from the theoretical value, indicating that the chains grew in a controlled manner. In a tubular reactor, given some residence time distribution attributable to laminar flow or axial dispersion, it is expected that the polydispersity of a controlled polymerization conducted

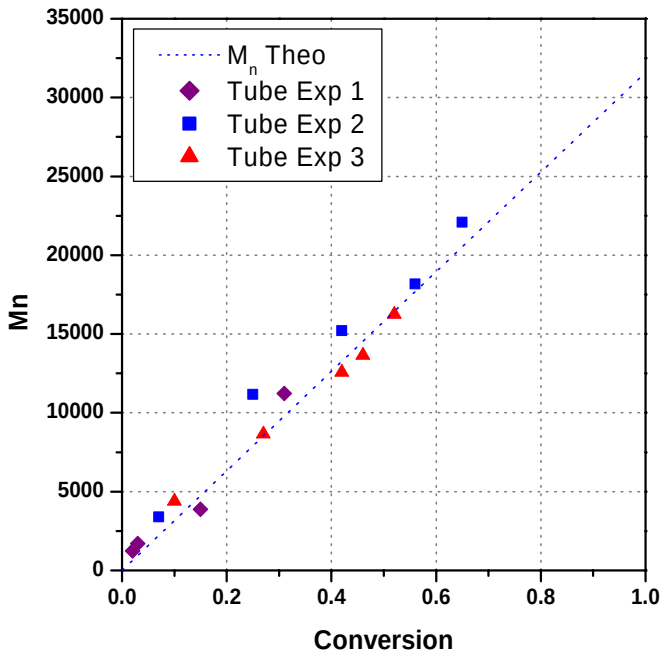


Figure 3.9 Plot of the number average molecular weight of polymers produced via the styrene/PEPDTA RAFT miniemulsion as a function of conversion.

in a tubular reactor will be adversely affected. As Figure 3.10 clearly indicates, in each case the polydispersities in the tube reactor were higher than those in batch. Since the batch and continuous polymerizations were initiated from the same unpolymerized miniemulsion, it is likely that the difference is primarily attributable to the effects of the flow regime in the tubular reactor. These effects will be quantified and discussed in the following chapter.

Another curious aspect of this system is shown (Figure 3.11a) in the GPC traces of the UV signal at 311 nm, reflecting the C=S of the dithioester, of experiment T2/B2. Because the UV signal at this wavelength is proportional to only the incorporated RAFT

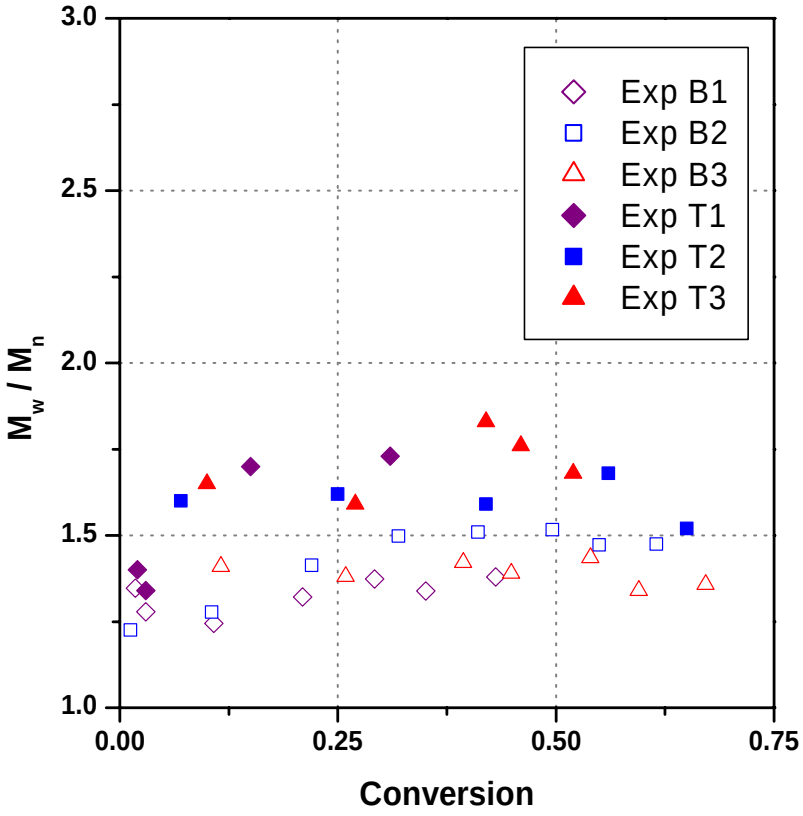


Figure 3.10 Plot of the polydispersities of styrene/PEPDTA RAFT miniemulsion in both the tubular and batch reactors.

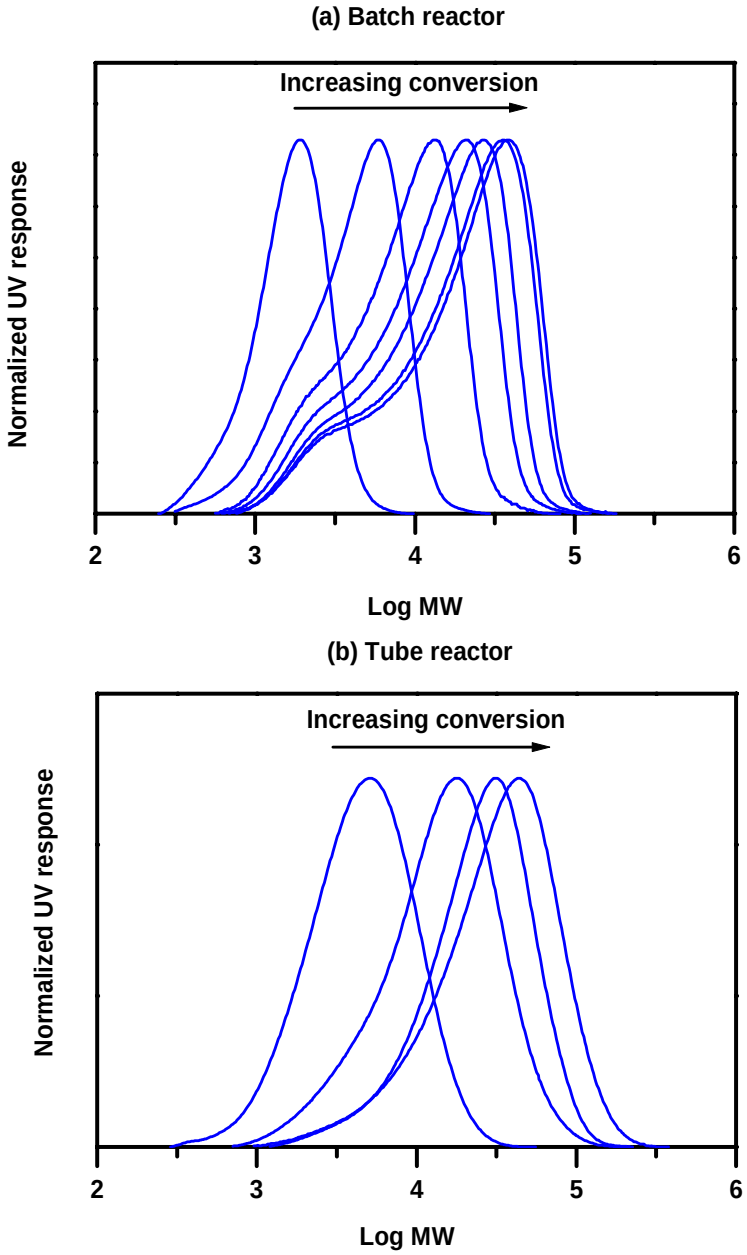


Figure 3.11 Evolution of the number distribution of chains using the normalized detector response of the UV absorbance at 311 nm which is indicative of the distribution of the dithioester RAFT agent within the polymer. Shown are the results from experiments T2 and B2.

agent, the traces reflect the number distribution, not the molecular weight distribution. It reveals an underlying population of non-growing chains, evidenced by an emerging short chain length peak that does increase in length with the reaction. The bimodality does not reveal itself as visibly in the GPC traces generated by the RI signal since these reflect the weight and not the number distribution. (See Figure 3.12) This is because the molecular weight distribution is recovered from the number distribution by multiplying the number distribution by the square of the molecular weight.[14] As such, the weight distribution will show some commensurate degree of broadening but it may necessarily not reveal the bimodality.

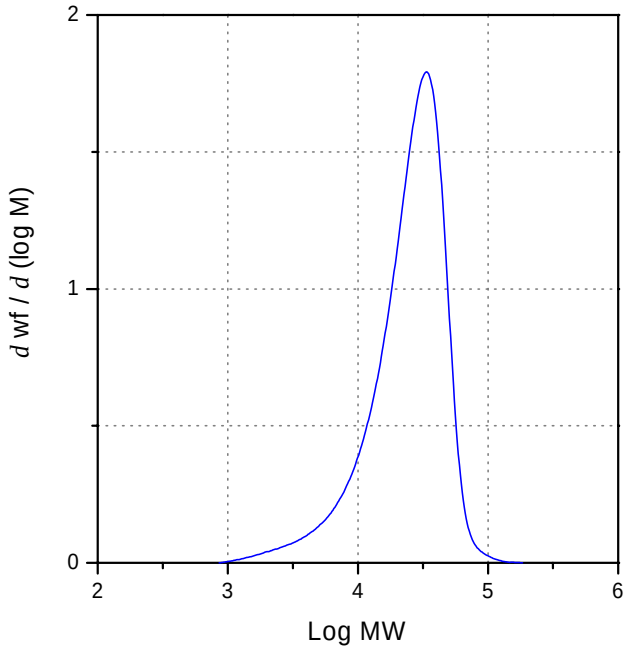


Figure 3.12 GPC trace from the RI signal, Exp B2, sample 7. Compare to UV trace for the same sample (rightmost chromatogram) in Figure 3.11a.

Since the chains still possess the dithioester moiety, it is very unlikely that they are dead chains. Rather, they appear to be dormant chains that for whatever reason no longer had access to monomer and therefore could no longer propagate. A similar phenomenon was observed by Smulders et al.[15] and was tested by soaking the dried latex in additional monomer and initiator and attempting to "grow" the chains. It was observed that by doing so the bimodality disappeared very quickly, incorporating itself into the overall growing chain population. This would indicate that the chains were in fact dormant and non-growing, not dead, and that some anomaly in the miniemulsion could be responsible for the observed bimodality. Further evidence for the livingness of these chains is revealed in the UV (311 nm) traces of a miniemulsion from the tube reactor that was used in a chain extension experiment using butyl acrylate. Figure 3.13 shows that the initial polystyrene contained a population of these non-growing chains. The final copolymer peak indicates that the chains shifted towards higher molecular weights.

One hypothesis as the cause for the observed bimodality is uneven droplet nucleation, in which case the earliest nucleated droplets would grow at the expense of monomer in droplets nucleated later. This uneven nucleation is relatively common in miniemulsions. If the effect were large enough, the early nucleated droplets could deplete the monomer in the late nucleated droplets to the extent that they would begin to propagate either very slowly or not at all. Another possible scenario is a population of very small droplets that in the presence of relatively even nucleation quickly deplete their supply of monomer. Observed at times in either of the two reaction systems, the phenomenon does not appear to be reactor-related. The non-growing chains can cause the

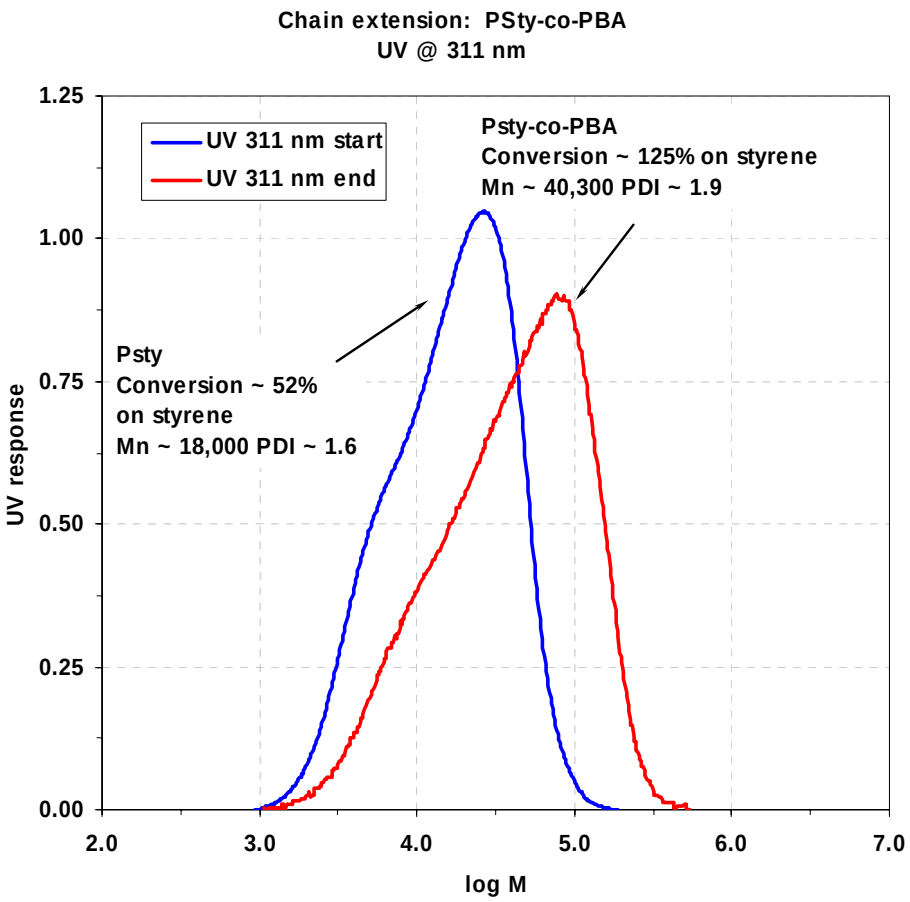


Figure 3.13 GPC UV traces at 311 nm reflecting the number distribution of the starting polystyrene and the final polystyrene-*co*-poly(*n*-butyl acrylate) produced by chain extension of the polymer latex from Exp T3.

polydispersity to increase, not decrease as expected, as the reaction progresses. The M_w/M_n evolution of Exp B2 (see Figure 3.10) is emblematic, clearly showing an increasing trend. What is significant is that even with the increased polydispersity of B2, the trend was still below that of T2, its tube analog experiment. In fact in each case, regardless of whether or not non-growing chains were observed, tube polydispersities

were consistently higher than batch, again pointing towards significant effects from a residence time distribution.

As mentioned earlier, a chain extension experiment was conducted with latex from the tubular reactor as a test of the "livingness" of the chains. Latex was taken from the longest residence time tube in Exp T3, (~ 52% conversion), and placed in a 3 neck reaction flask outfitted with septum, N_2 purge, and condenser. A second monomer, *n*-butyl acrylate was added to the flask, the miniemulsion was allowed to purge for ~30 minutes, then placed in a heated bath. No additional initiator was added to the system. Since butyl acrylate is invisible to UV at 254 nm, the GPC UV trace can be used with this system to verify incorporation of this second monomer. This is shown in Figure 3.14, where a shift in the signal towards high molecular weights combined with >100% conversion on based on the original mass of styrene indicates growth of the chains and confirms integration of the second monomer and formation of copolymer. Additionally, since the UV signal and RI signal essentially overlap, the copolymer formed was very pure, i.e. with little or no formation of homopolymer.

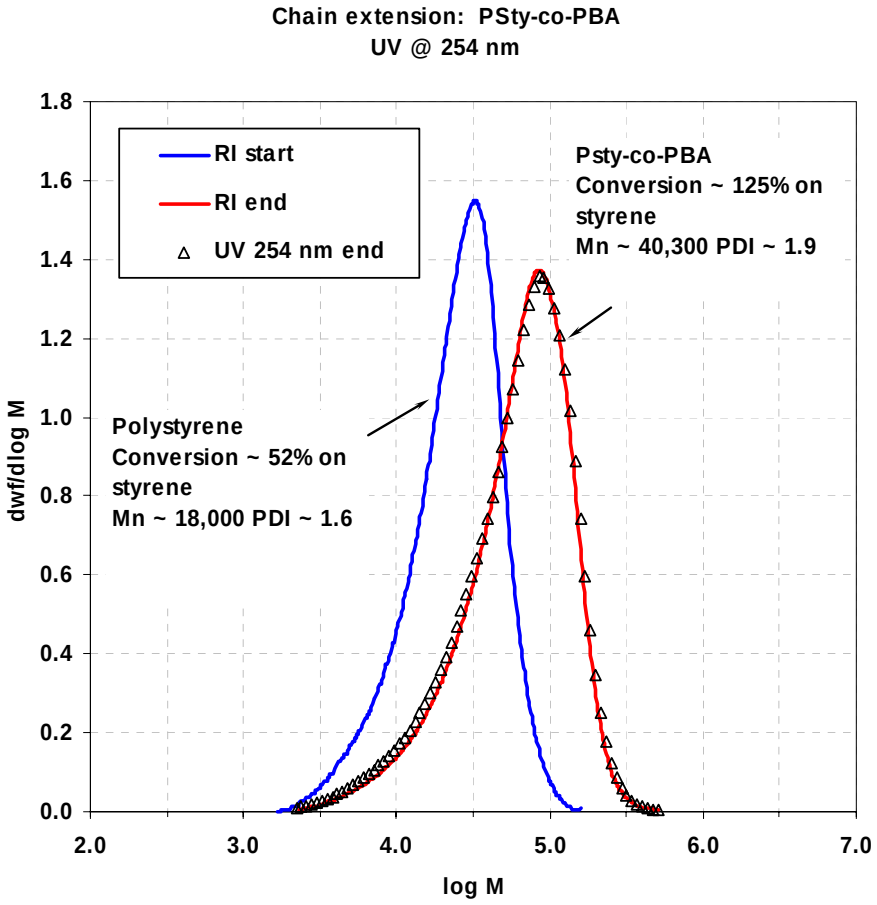


Figure 3.14 GPC traces of starting polystyrene, RI only, and final copolymer, polystyrene-*co*-poly(*n*-butyl acrylate), RI and UV at 254 nm, produced by chain extension of the polymer latex from Exp T3.

3.3 Conclusions

In this chapter the use of the RAFT polymerization technique combined with miniemulsion in a tubular reaction system has been explored in some detail. Recipes were developed in batch using a dual surfactant system to promote stable latexes. It was demonstrated that a 1/1 molar SDS/Triton X-405 surfactant system provided excellent stability of the miniemulsions, in both tube and batch, with no visible latex separation or coagulum. This is attributed to greater total surface coverage of the surfactants on the droplets when employed in tandem. Some apparent inhibition was observed in styrene polymerizations with PEPDTA and preferential addition of primary radicals to the RAFT agent was offered as the most likely source. Both experimental data and literature references were utilized to buttress this hypothesis. Experiments in a continuous tubular reaction system revealed similar kinetics to batch, however in each case the reaction progressed at a slightly faster rate in the tube reactor. The error in flow rate was quantified and, taken in conjunction with the steady state reactor profile, was eliminated as a possible source of the advanced rates. Slight temperature differences between the batch and tube were offered as a possible cause of the variation, however, calculations of initiator decomposition at the particle sizes and reaction parameters used here effectively ruled out this suggestion. Small differences in the particle size could cause the discrepancy, and the data presented here, while inconclusive, can not positively rule out the possibility. The polydispersity of the polymer produced in the tube reactor was consistently higher than that produced in concurrent batch experiments, which suggests that a residence time distribution of the droplets/particles played a contributing role.

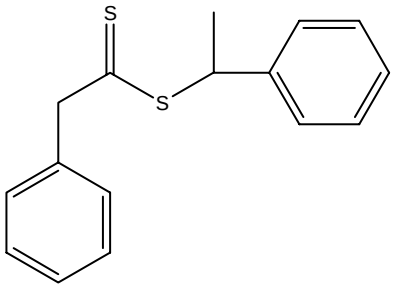
An underlying population of non-growing chains which served to broaden the molecular weight distribution was identified at times in both the batch and tube polymer. It was demonstrated experimentally that the chains were indeed non-growing, dormant chains and not simply dead chains. Uneven droplet nucleation was postulated as the cause, leaving some particles without sufficient monomer to propagate. However, further work will need to be done in order to positively identify this as the culprit. Finally, it was shown through chain extension that the polymer produced in the tube retained its controlled character to produce copolymer of polystyrene-co-poly(*n*-butyl acrylate), opening the possibility of employing the system in the production of block copolymers.

3.4 Experimental

3.4.1 Materials

Styrene and butyl acrylate (monomers, ≥ 99.0%, Aldrich) were cleaned by either vacuum distillation or by running through a column packed with inhibitor remover. The column packing was purchased from Aldrich and was specific to the type of inhibitor in the monomer. Potassium persulfate (initiator, KPS, ≥99.0 %, Aldrich), Triton X-405 (non-ionic surfactant, TX405, 70% solution in water, Aldrich), hexadecane (co-stabilizer, ≥ 99.0%, Aldrich), and sodium dodecyl sulfate (ionic surfactant, SDS, ≥99%, Aldrich) were used as received. The reagents for the RAFT agent synthesis, hydrochloric acid (37% in H2O, Aldrich), magnesium sulfate (≥ 99%, Aldrich), p-toluenesulfonic acid (≥ 98%, Aldrich), benzyl chloride (≥ 99.9%, J.T. Baker), carbon disulfide (≥ 99.9%, J.T. Baker), carbon tetrachloride (≥ 99.9%, Aldrich), anhydrous diethyl ether (≥ 99.9%, Fisher) and magnesium turnings (≥ 98.0%, Aldrich) were used as received. Deionized water was generated in-house with a U.S. Filter Systems Deionizer and was used without further purification.

3.4.2 Synthesis of RAFT agent



Scheme 3.1 RAFT agent used in this study

Synthesis of 1-phenylethyl phenyldithioacetate (PEPDTA, see Scheme 3.1)[16] Benzyl chloride (88.0 g, 0.69 mol) was added dropwise over a period of 4 hours under ultra-high purity nitrogen to magnesium turnings (15.1 g, 0.62 mol) in anhydrous diethyl ether (400 mL). The mixture was stirred and kept on ice. After a thick, gray suspension formed indicating the initial reaction, the mixture was heated to 34 ºC and allowed to reflux for 2 hours. The mixture was then chilled using an ice bath and additional diethyl ether (200 mL) was introduced. Carbon disulfide (50.0 g, 0.66 mol) was then added dropwise over a 2 hour period. A golden, yellow suspension formed immediately and was stirred for an additional 2 hours after complete introduction of the carbon disulfide. The mixture was then poured into ice water (1200 mL) and the aqueous portion collected following three washes with clean ether. After adding a final layer of ether, the mixture was acidified with 37% aqueous hydrochloric acid. The product, phenyldithioacetic acid, (~ 30.5 g, 0.17 mol, 25%) was collected following rotary evaporation of the ether, filtering with magnesium sulfate to remove any residual moisture, and then placing under high vacuum for 1 hour to remove any traces of ether. The phenyldithioacetic acid was then reacted with styrene (19.0 g, 0.18 mol) and a trace amount of p-toluenesulfonic acid (~ 0.05 g) as catalyst in carbon tetrachloride (~50 mL) at 70 ºC. The mixture was refluxed under ultra-high purity argon for 18 hours. After removal of the carbon tetrachloride via rotary evaporation, the product was precipitated and recrystallized from cold methanol as bright, yellow crystals and placed under high vacuum overnight (~ 23.0 g, 0.09 mol, 13% on benzyl chloride). ^{1}H NMR (CDCL3): δ (ppm) 1.7 d (3H), 4.2 s (2H), 5.1 q (1H), 7.3 m (10H).

3.4.3 Batch miniemulsion equipment and procedure

All of the batch miniemulsion reactions except those run concurrent to a continuous reaction were prepared and carried out using the following procedure. The surfactants, SDS and TX405, were first added to water and allowed to mix for 15 minutes. The monomer, hexadecane, and RAFT agent were combined and allowed to mix for 15 minutes. The organic phase was then added to the aqueous phase and agitated vigorously with a magnetic stirrer for 30 minutes, forming a light yellow emulsion. The miniemulsion was formed by sonicating for 20 minutes (Fisher 300 Sonic Dismembrator at 70% output). During the sonication, the miniemulsion was cooled by an ice bath in order to keep the temperature low so that any thermal initiation of the monomer would be minimized. After sonication, the miniemulsion was transferred to a 100 mL, round-bottomed, 4-neck flask outfitted with a septum, reflux condenser, nitrogen feed and thermometer to monitor the temperature of the miniemulsion. The miniemulsion was kept agitated by a magnetic stirrer. After allowing the miniemulsion to de-oxygenate under ultra high purity nitrogen for 30 minutes, the flask was immersed in an oil bath that had been preheated to the desired reaction temperature. The temperature of the miniemulsion was monitored with the thermometer and when it had reached the reaction temperature, a solution of initiator and water was injected through the septum. The time of initiator injection was considered time zero and samples were withdrawn through the septum via syringe at regular intervals for conversion, GPC, and particle size analysis. The reaction was kept under nitrogen for the entire time of the experiment.

3.4.4 Continuous miniemulsion equipment and procedure

A schematic illustration of the reaction system is shown in Figure 3.15. The reactor was constructed of five separate 1/8" OD - 1/16" ID PFA (perfluoroalkoxy, a copolymer of TFE) tubes with different lengths in order to turn out five different residence times. The transparency of the tubing facilitated inspection for plugging and fouling. The tubes were arranged in 25 cm diameter helical coils and submerged in a constant temperature, 75 liter water bath. Lengths varied from 7.6 meters to 38.1 meters in 7.6 meter increments, with the tubes numbered 1 through 5, respectively. The temperature of the bath was controlled with a VWR 1122S immersion circulator. Because of the high heat transfer inherent in tube reactors and the small size of the tubing in relation to the water bath, the temperature was assumed to be constant throughout the length of the submerged tubing.

Two peristaltic tubing pumps were utilized to feed the miniemulsion into the five tubes of the reactor. The drives were Masterflex® variable speed (1-100 rpm) console drives outfitted to accommodate multiple pump heads. Tubes 1-3 were supplied via three separate pump heads mounted on the first drive and tubes 4 and 5 were supplied by two separate pump heads mounted on the second drive. The pump heads were Masterflex® L/S® Standard pump heads (polycarbonate housing with SS rollers) and equipped with L/S® 13 Viton® tubing. The combination of drive, head and tubing was rated to allow flow rates from 0.06 to 6.0 mL/min. The pumps were calibrated beforehand using deionized water and by measuring the time to fill a 5 mL volumetric flask at four different pump settings. During the actual experiment flows were checked two ways, both by measuring the time for the miniemulsion to traverse a specified length of tubing and

volumetrically as before. Measurements were taken throughout the experiment and it was observed that flowrates remained constant (within ± 0.01 mL/min), indicating that swelling of the Viton® tubing was negligible if existent at all. The miniemulsion was prepared by separately mixing the organics and aqueous components, then placing all the ingredients except the initiator into the main feed tank, a 3-L round, three neck (center 45/50, sides 24/40) flask, with heavy agitation provided by a Talboys Engineering Model 102 laboratory stirrer outfitted with a glass shaft and Teflon® paddle. The emulsion formed by the heavy agitation was kept under ultra high purity nitrogen and allowed to de-oxygenate overnight. The miniemulsion was produced by pumping the emulsion into a continuous sonication vessel, where it was stirred with a magnetic stirrer. The miniemulsion could be then fed to the tubular reactor on a continuous basis. The sonication was provided by a Fisher 300 sonic dismembrator set at 80% output. Two reciprocating pumps were used for the continuous sonication loop. An FMI QG 50 laboratory pump supplied the sonication vessel with feed emulsion. The volume (and average residence time of the forming miniemulsion) in the sonication vessel was kept constant via pumping off excess liquid and recycling into the feed tank with an FMI QG 20 laboratory pump. Both pumps were outfitted with Kynar® pump heads with ¼-inch SS pistons and carbon liners. The average residence time in the sonication vessel was ~30 minutes.

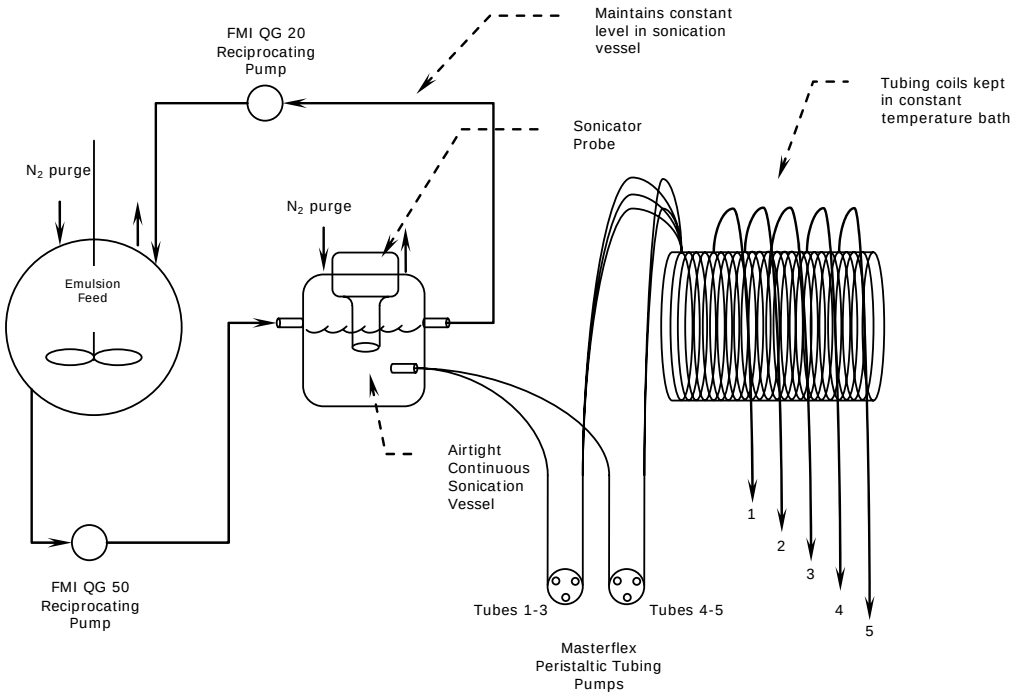


Figure 3.15 Schematic illustration of the tubular reactor for RAFT polymerizations in continuous miniemulsion.

In order to run a concurrent batch experiment, an initial miniemulsion charge without initiator was bled off through the sonication vessel. Initiator was added to the batch after it had been allowed to reach reaction temperature. To begin the tubular experiments, after bleeding off a charge for the batch experiment, sonication was stopped, initiator was added to the feed tank and the mixture was allowed to mix with the emulsion for 1 hour. The tube reactor was then started by again pumping the emulsion into the sonication vessel, sonicating, and sending the formed miniemulsion to the reactor. All of the tubes were initially filled with water. In order to suppress initiation before the feed entered the

reactor, the feed tank and sonication vessel were kept at 5 ºC via submersion in a refrigerated water/ethylene glycol bath. Refrigeration and circulation were supplied by a VWR 1186D 28L programmable heating/cooling circulating bath. Samples of the feed were taken throughout the experiment and analyzed gravimetrically for conversion. The analysis indicated no conversion in the feed during the experiment, with a measurement error margin of ± 2% conversion.

3.4.5 Characterization

Polymer latex samples were dried for 24 hours in a vacuum oven (50 ºC, ~ 100 kPa vacuum) and monomer conversion was subsequently determined gravimetrically. Samples were dissolved in THF and run through an alumina pipette column to remove the TX405 polymeric surfactant. The number average molecular weight, M_n, and the polydispersity, M_w/M_n, were calculated using data gathered via size exclusion chromatography in THF. Three columns (American Polymer Standards styrene-divinylbenzene 100 Å, 1000 Å, and 105 Å) mounted in a Waters WAT038040 column heater set at 30 ºC were utilized. The columns were connected to a Viscotek GPCMax pump/autoinjector, a Waters 410 refractive index detector, an LDC Milton Roy Spectromonitor 3000 UV detector (at either 254 nm or 311 nm) and calibrated against 10 narrow polystyrene standards (Polymer Laboratories; M_n = 580-200K, M_w/M_n = 1.02-1.16). Latex particle sizes and polydispersities were analyzed using quasi-elastic light scattering (QELS, Protein Solutions DynaPro99 with DynaPro DCS v 5.26 software).

3.5 References

1. Russum, J. P.; Jones, C. W.; Schork, F. J. *Ind. Eng. Chem. Res.* **2005**, *44*, 2484.

2. Russum, J. P.; Jones, C. W.; Schork, F. J. *Macromol. Rapid Comm.* **2004**, *25*, 1064.

3. Levenspiel, O., *Chemical Reaction Engineering*. 2d ed.; Wiley: New York, NY, 1972.

4. Lansalot, M.; Davis, T. P.; Heuts, J. P. A. *Macromolecules* **2002**, *35*, 7582.

5. Smulders, W. Macromolecular Architecture in Aqueous Dispersions. Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, 2002.

6. de Brouwer, H.; Tsavalas, J. G.; Schork, F. J.; Monteiro, M. J. *Macromolecules* **2000**, *33*, 9239.

7. Moad, G.; Chiefari, J.; Chong, Y. K.; Krstina, J.; Mayadunne, R. T. A.; Postma, A.; Rizzardo, E.; Thang, S. H. *Polym. Int.* **2000**, *49*, 993.

8. Colombié, D.; Landfester, K.; Sudol, E. D.; El-Aasser, M. S. *Langmuir* **2000**, *16*, 7905.

9. Goto, A.; Sato, K.; Tsujii, Y.; Fukuda, T.; Moad, G.; Rizzardo, E.; Thang, S. H. *Macromolecules* **2001**, *34*, 402.

10. Wang, A. R.; Zhu, S. P. *J. Polym. Sci. Polym. Chem.* **2003**, *41*, 1553.

11. McLeary, J. B.; Calitz, F. M.; McKenzie, J. M.; Tonge, M. P.; Sanderson, R. D.; Klumperman, B. *Macromolecules* **2004**, *37*, 2383.

12. McLeary, J. B.; McKenzie, J. M.; Tonge, M. P.; Sanderson, R. D.; Klumperman, B. *Chem. Commun.* **2004**, 1950.

13. Jódar-Reyes, A. B.; Ortega-Vinuesa, J. L.; Martín-Rodríguez, A.; Leermakers, F. A. M. *Langmuir* **2002**, *18*, 8706.

14. Gilbert, R. G., *Emulsion Polymerization: A Mechanistic Approach*. Academic Press: 1995.

15. Smulders, W. W.; Jones, C. W.; Schork, F. J. *AIChE J.* **2005**, *51*, 1009.

16. Quinn, J. F.; Rizzardo, E.; Davis, T. P. *Chem. Commun.* **2001**, 1044.

CHAPTER 4

IMPACT OF FLOW REGIME ON POLYDISPERSITY IN TUBULAR RAFT MINIEMULSION POLYMERIZATION[†]

4.1 Introduction

This chapter expands upon the previous studies of styrene RAFT/miniemulsion polymerization in a continuous tubular reactor.[1,2] The focus here will be on the flow characteristics of the reactor in an attempt to answer the second question posed in the introduction to Chapter 3. That is, what is the effect of laminar flow or of significant axial dispersion on the final polydispersity? A mathematical model is presented in order to understand the general effects of a laminar flow regime on the conversion and polydispersity of the polymer formed using the RAFT mechanism. The flow characteristics of the reactor are quantified utilizing a modified dye tracer technique developed exclusively for this type of heterogeneous system. Isolated plug flow was achieved using metered N_2, effectively producing an ideal flow regime. This allowed for the study of the effect of the residence time distribution (RTD) on the polydispersity (PDI) of the polymer produced. Finally, an unexpected transient behavior of the reactor during startup of isolated plug flow was observed and is explained using empirical data from solution experiments and a mathematical model.

[†] Portions of this chapter have been accepted for publication and are currently in press, *AIChE J.*

4.2 Results and Discussion

4.2.1 Modeling RAFT chemistry in laminar flow

It should be stated at the outset that it is not the purpose of this section to present a new or even improved mathematical model of RAFT chemistry. Rather, the purpose here is to take up an aspect of RAFT chemistry that has not been heretofore addressed in the open literature, and that is the behavior of these systems in cylindrical laminar flow. As pointed out in Chapter 3, the Reynolds number of the first series of experiments was very low, ~ 10-20. Taken alone, this would indicate that the reactor was operating in the laminar flow regime. If this were the case, then a residence time distribution (RTD) would be imparted on the miniemulsion particles and in turn have a detrimental effect on the polydispersity of the polymer formed. As the data show, the PDI of the polymer formed in the tube was indeed higher than that formed in batch. The question to be asked is whether the difference can be attributed to the RTD inherent to laminar flow. In an effort to understand just how much influence the laminar flow RTD has on PDI, a suitable existent RAFT model was selected and applied to a tubular reactor in laminar flow. So that the results are understood in their proper context, a brief synopsis of the three most accepted existing models and their mechanistic underpinnings will first be presented, followed by the rationale for choosing a particular model. Finally, the results will be compared with kinetic and polydispersity data presented in Chapter 3 an effort to understand whether or not the differences observed, particularly in the polydispersity, between the actual batch and tube could be explained in terms of laminar flow.

As alluded to previously, there exists in the open literature a number of published studies addressing in some form or fashion the modeling of the fundamental RAFT

process.[3-19] Of these, two apply high level ab initio molecular orbital calculations in an attempt determine the forward and reverse rate coefficients for a series of RAFT agents, [8,9] one utilizes a Monte Carlo method to model the kinetics,[4] two develop models applying the method of moments to the fundamental differential equations of the kinetics,[18,19] and one develops a model by simplifying the pre-equilibrium reaction to one irreversible chain transfer reaction, using a boundary density integral approach to simplify the chain populations in the main equilibrium, and employing the PREDICI® software package to perform the numerical analysis.[3] The remaining works employ one of these techniques, either as presented in the original work or by extending upon it, in an attempt to shed light on one aspect or another of the process. Of the fundamental studies mentioned, only one, the work of Zhang and Ray,[18] applies the model to continuous reaction systems. They simulated RAFT chemistry in a semibatch reactor, a single continuous stirred reactor (CSTR) and a series of stirred reactors. In studying the effects of feed rates, initiator concentrations, initiator half-lives, RAFT agent concentrations and RAFT agent equilibrium constants, they concluded that both semibatch systems and a series of CSTRs could both be successfully employed to produce controlled polymer architectures. The results are instructive in that they highlight the necessity for a judicious selection of reactants and the need for further understanding of the mechanism and the interplay between the various rate constants. No two systems can be considered remotely similar in this regard and changing any one component (monomer, initiator, RAFT agent) can have profound effects upon the performance of the reaction. Indeed, all of the models so far presented highlight this necessity. While the state of the art is evolving and still has some way to go before truly predictive, quantitative models emerge, what currently exists

does help at least to shed some light on the intricate workings of RAFT systems. With the exception of one work,[14] all of the models currently exsistent in the published literature apply to bulk or solution polymerizations, not to dispersed systems like emulsions or miniemulsions.

The complexities of the mechanism and the lack of reliable kinetic parameters have presented enormous challenges to producing accurate and predictive models. With regard to the mechanism, one of the problems has been fully accounting for what can happen during the important pre-equilibrium phase of the reaction (see Chapter 1, Scheme 1.1). Initially, it was simply assumed that the RAFT agent incorporated very quickly, i.e. C_T is very high, and the initial pre-equilibrium step could be neglected. Adding to the difficulty is the fact that rate constants for the forward and reverse reaction and any intermediate termination reactions were simply not available. While progress is being made in this regard, much remains to be understood. Related to this, an ongoing vigorous and friendly debate exists in the scientific community with regard to the rate retardation that is often seen in these systems.[20,21] This presents a second source of uncertainty. The argument here centers on the intermediate radicals (Scheme 1.1, **5**) and whether the retardation is caused by their slow fragmentation[3,8,15-17,20,22] or simply by their termination (either reversibly[23,24] or irreversibly[25,26]) with other radicals in the system.

In the effort presented here, the method of moments was selected because it is familiar and can be freely implemented using FORTRAN, as opposed to the aforementioned PREDICI® package. The RAFT model is based on the work of Zhang and Ray mentioned earlier. A detailed treatment will not be presented here, the reader is referred to their work for an excellent description of the equations involved. While Wang

and Zhu[19] later employed this technique to model the system, adding the cross-termination product of the intermediate radical (Scheme 1.1, **3** and **5**) to form three-armed chains, the simpler Zhang model was selected because of the lack of good parameters for the cross-termination reaction and the fact the model was validated using experimental data.

Because no RTD exists in ideal, or what is often referred to as "plug"[27] or "piston"[28] flow, a tubular reactor will behave in an identical manner as a batch reactor. One way this can be achieved is by insuring that the Reynolds number is high enough (> 4000) so that the flow regime is turbulent. As such, any model describing a batch reactor would also be applicable to a tube reactor in ideal flow. However, in practice achieving ideal flow can be difficult and in many, if not most, instances real tube reactors operate in a non-ideal flow regime with some commensurate RTD. Of the three model reports[18,29,30] in the open literature applying controlled or "living" systems to continuous systems, only one studied the effects of non-ideal flow on polymer properties.[30] An axial dispersion model was applied to the atom transfer radical copolymerization of styrene and *n*-butyl acrylate in a tubular reactor, but the effects of laminar flow were not studied. The two remaining reports[18,29] stop short of non-ideal flow, simply reporting the case for a batch reactor and pointing out the fact that kinetically an ideal tubular reactor would behave in the same way.

For the general case of isothermal laminar flow in a circular tube with a parabolic velocity profile and complete segregation, the following relation between batch and tube applies[28]:

$$f_{out} = \int_{\bar{t}/2}^{\infty} f_{batch}(t)\frac{\bar{t}^2}{2t^3}dt \qquad (4.1)$$

This relationship can be applied to any reaction and the function $f_{batch}(t)$ can represent any property of the reaction that is a function of time, even a complex reaction like RAFT whose resolution lies in solving a system of many ODEs. In this work, $f_{batch}(t)$ would represent one of the moments of the chain distribution.

The representative system modeled here is a styrene/PEPDTA bulk RAFT polymerization with AIBN as initiator. The parameters are outlined in Table 4.1. The temperature was assumed to be 60 °C and all the rate constants except those of PEPDTA were calculated using that temperature. The forward and reverse rate constants for PEPDTA are estimates at 60 °C taken from Lansalot et al.[31] There is no accounting for bulk fluid viscosity effects as conversion increases because they would be negligible in a miniemulsion reaction. In other words, each particle would behave as an independent mini-batch/bulk reactor traveling a particular streamline of the laminar flow velocity distribution.

Table 4.1 RAFT in laminar flow model parameters

Quantity	Value	Units	Ref
$[M]_0$	8.865	mol/L	
$[RAFT]_0$	1.12×10^{-2}	mol/L	
$[I]_0$	1.12×10^{-2}	mol/L	
k_{add}	5.6×10^{5}	$L \cdot mol^{-1} \cdot s^{-1}$	[31]
k_{-add}	2.7×10^{-1}	s^{-1}	[31]
k_d	1.3×10^{-5}	s^{-1}	[32]
k_p	340	$L \cdot mol^{-1} \cdot s^{-1}$	[33]
k_t	5.5×10^{7}	$L \cdot mol^{-1} \cdot s^{-1}$	[34]
k_{trm}	6×10^{-5}	$L \cdot mol^{-1} \cdot s^{-1}$	[32]
f	0.65		

Results of the simulation are shown in Figure 4.1. They suggest that the rate of polymerization, while slower in the tube, is not as significantly affected as the polydispersity. For both the conversion and PDI, the effects are especially pronounced at conversions below ~ 50%. As the residence time (and therefore the conversion) increases, the variance between batch and tube tends to even out and both conversion and PDI approach batch values. This indicates that it may be possible to minimize the significant negative impact on PDI by simply insuring that the conversion is sufficiently high. Of course, this is a general statement and each system would have to be considered individually in order to access exactly what conversion would be sufficiently high.

With regard to the polydispersity data presented in Chapter 3 (see Figure 3.10), it is difficult to draw any absolute conclusions when comparing the data to the simulation since the experimental conditions were not the same. (This was necessitated primarily by the rate constants that were available for the RAFT agent.) While the experimental PDIs do trend downwards towards the batch values as the reaction progresses, the variance early in the reaction appears not to be as wide as that suggested by the model. Because of this it is not possible to say with any certainty that the higher PDIs are attributable primarily to laminar flow in the tube reactor. What needs to be understood is the actual flow profile in the reactor, which is the subject of the next section.

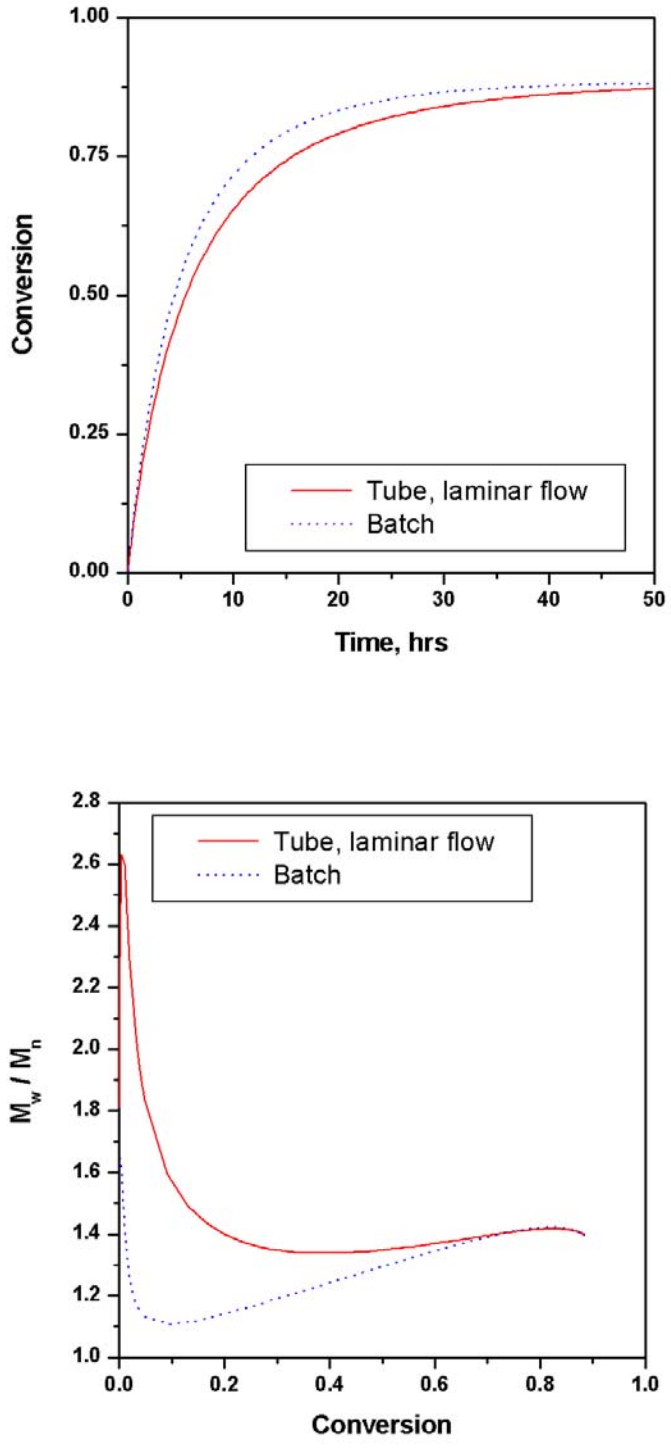


Figure 4.1 Effects of laminar flow on monomer conversion and polydispersity for RAFT polymerization of styrene

4.2.2 Modified tracer experiments

Traditionally, flow behavior in continuous reactors has been most often determined using dye or electrolyte injection techniques. The solute is injected, usually as either an impulse or step input, at some known time and the concentration is monitored at the outlet of the reactor. The concentration profile as a function of time reveals the flow characteristics of the reactor, its deviation from non-ideal flow. However, this classical method assumes a homogeneous reaction media, a solution for example. Since miniemulsions are heterogeneous dispersions of droplets/particles in water, this technique requires modification in order to develop the correct flow profile for a given reactor. Since the polymerization takes place in the small, dispersed particles one needs information regarding their residence time, not that of the bulk fluid. This can be achieved in principle by simply adding an extremely hydrophobic dye to the monomer droplets. By injecting this dye-containing miniemulsion into the feed stream, a more accurate picture of the flow characteristics of the reactor during miniemulsion polymerizations is developed. In this work, and oil soluble dye (*N*-ethyl-*N*-(2-hydroxyethyl)-4-(4-nitrophenylazo)aniline, Disperse Red 1) was utilized. It was selected not only for its hydrophobicity but also for its maximum or peak absorbance, $\lambda_{max} \sim 503$ nm, which is much higher than the absorbance of any of the system components. This is important because the UV response of the effluent should only derive from the dye for the results to be useful. This was confirmed by SEC/UV of the UV active system components, styrene and TX405.

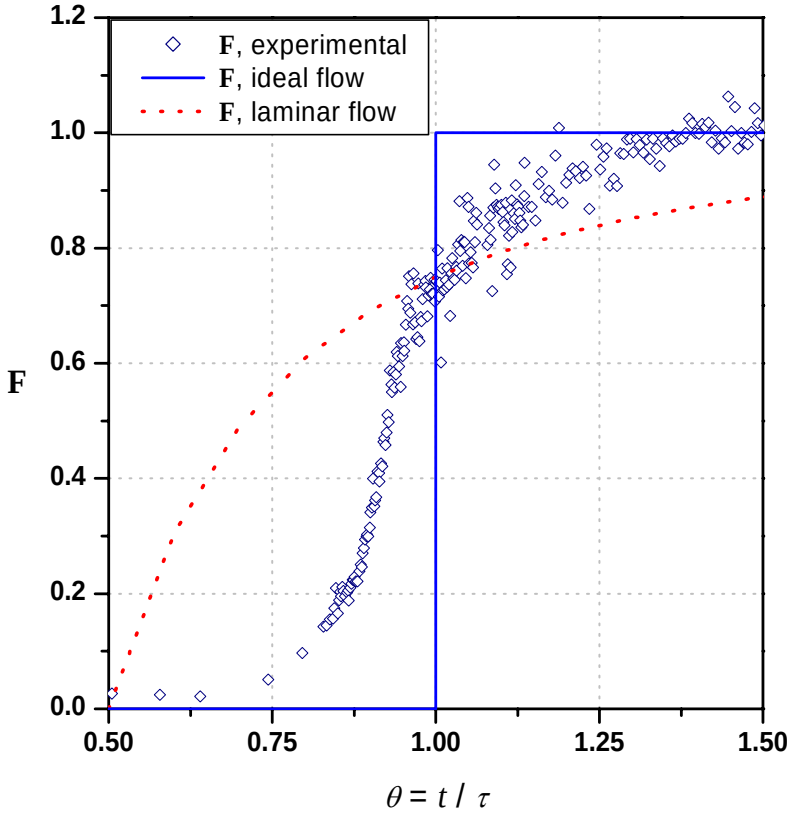


Figure 4.2 Experimental and theoretical step response curves for miniemulsion in tube reactor, $R_e \sim 20$.

Figure 4.2 shows the results of the tracer experiment. The dotted line represents the theoretical **F**-curve for a tubular reactor in laminar flow: [27]

$$\mathbf{F} = 1 - \frac{1}{4\theta^2} \quad \text{for} \quad \theta \geq \frac{1}{2} \quad \text{where} \quad \theta = t/\tau \qquad (4.2)$$

The data reveal that the flow profile in the reactor deviates from the ideal. The degree to which a given flow exhibits non-ideal behavior is frequently quantified by use

of the dispersion model, an analog to Fick's law of diffusion in a cylinder. In dimensionless form it is represented by:

$$\frac{\partial C}{\partial \theta} = \left(\frac{\mathbf{D}}{uL}\right)\frac{\partial^2 C}{\partial \zeta^2} - \frac{\partial C}{\partial \zeta} \qquad (4.3)$$

where

$$\zeta = (ut + z)/L \quad \text{and} \quad \theta = t/\tau = tu/L$$

and the dimensionless group $\mathbf{D}/uL$ is called the vessel dispersion number. It is this parameter that is commonly used as a measure of the degree axial dispersion in a tubular reactor. As $\mathbf{D}/uL$ approaches zero, the flow profile approaches ideal. Conversely, as $\mathbf{D}/uL$ increases, the flow profile becomes less ideal, approaching mixed flow. At the extremes:

$$\frac{\mathbf{D}}{uL} \to 0 \quad \text{ideal (plug) flow}$$
$$\frac{\mathbf{D}}{uL} \to \infty \quad \text{mixed flow}$$

Provided deviations from ideal flow are small, the RTD curve is symmetric and analytic solutions are available to (4.2), allowing easy determination of $\mathbf{D}/uL$ from the tracer curve. For larger deviations, when the RTD curve is nonsymmetrical, analytic expressions are not available and numerical methods must be used to calculate $\mathbf{D}/uL$. A close examination of the data presented in Figure 4.2 reveals that in this case the step response is nonsymmetrical, indicating a large deviation from ideal flow. In order to

quantify the degree of non-ideality and the vessel dispersion number, the step response curve must first be converted to a pulse response curve. They are related by:

$$\frac{d\mathbf{F}}{dt} = \mathbf{E} \qquad (4.4)$$

where **E** represents the pulse response curve. Here, the data were fitted using a smoothing spline,[35] which is shown below in Figure 4.3:

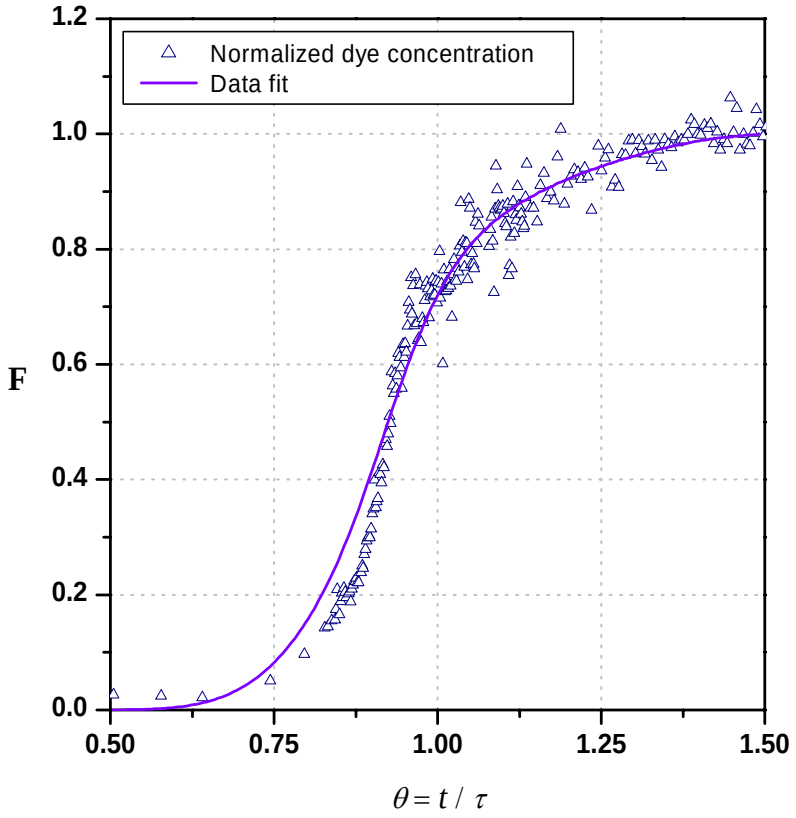


Figure 4.3 Original step response data and fit using a cubic smoothing spline, smoothing parameter $p = 0.9965$, $R^2 = 0.9683$

The data are then differentiated numerically using a central difference formula (4.5)

$$\frac{df}{dx} \approx \frac{f(x+h) - f(x-h)}{2h} \qquad (4.5)$$

and (4.4) to yield the necessary RTD curve, shown in Figure 4.4. As a point of reference, curves representing small and large deviations from ideal flow are shown. The non-ideality of the flow profile in the reactor is apparent when comparing the response curve generated from the data to that for near ideal flow. The reactor curve is clearly nonsymmetrical and skewed to the left, as would be expected as the flow deviates from ideal and approaches mixed flow. The extent of the deviation can be determined by examining the variance of the curve using:[36]

$$\sigma_\theta^2 = 2\left(\frac{\mathbf{D}}{uL}\right) - 2\left(\frac{\mathbf{D}}{uL}\right)^2 \left[1 - e^{-uL/\mathbf{D}}\right] \qquad (4.6)$$

It should be noted that this expression was derived utilizing closed boundary conditions. This "closed vessel" model is employed based upon the experimental conditions where the flow pattern is expected to change at the point of injection as well as at the outlet of the tube where the samples were collected. The vessel dispersion number was calculated to be ~ 0.10, an order of magnitude larger than what is normally considered to represent a small deviation ($\mathbf{D}/uL < 0.01$).

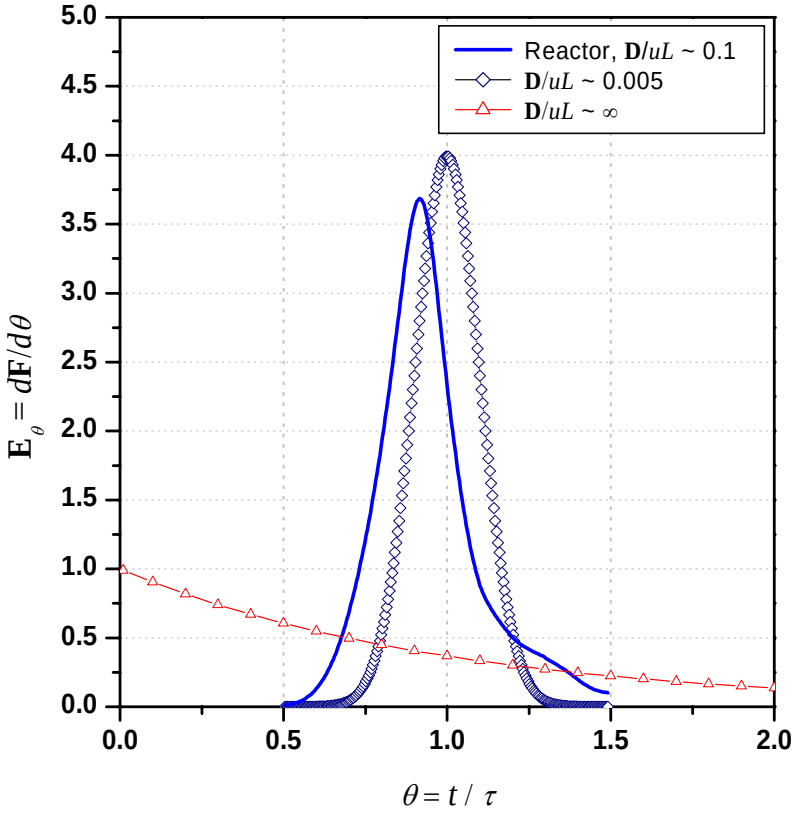


Figure 4.4 Pulse response curve ($\mathbf{E}_\theta$ curve) calculated from step resonse data. The two other curves compare the actual flow profile to curves that represent very small deviation from ideal flow ($\mathbf{D}/uL \sim 0.005$) and mixed flow ($\mathbf{D}/uL \sim \infty$)

Interestingly, even though the Reynolds number was very low (~ 20), the flow regime, while clearly non-ideal, is not laminar. Considering the low Reynolds number, this result was unanticipated. It can be explained in part by noting that in order to have fully developed laminar flow, a no-slip condition is required at the wall of the tube. In this case, the use of PFA tubing in combination with an aqueous dispersed system contributes to considerable slippage at the wall (See Figure 4.5). The surface tension of the aqueous phase is too high for the water to wet the wall of the PFA tube to any large degree.

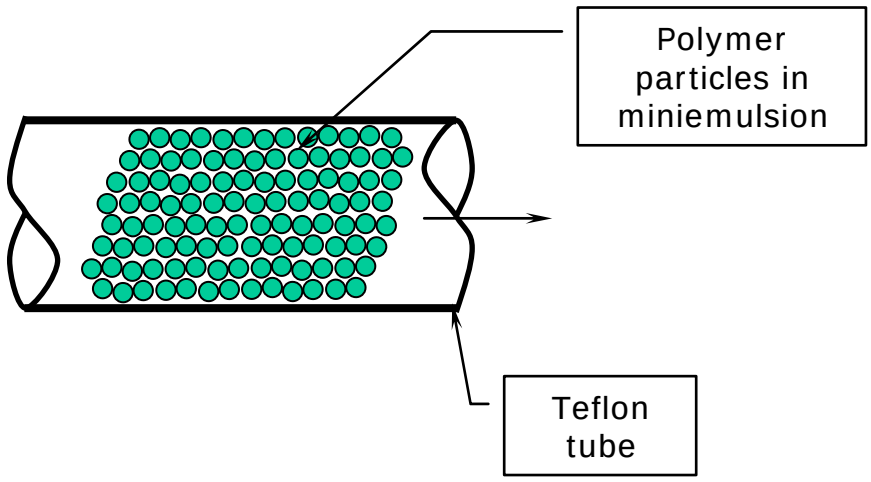


Figure 4.5 Graphic representation of miniemulsion flowing through tube with slippage at the tube wall

This conclusion is based also upon the physical observation that isolated plugs of miniemulsion flowing through the reactor did so without the considerable tailing that would be characteristic of no-slip conditions. The fact that the flow profile is non-ideal

suggests that diffusive effects arising from the ability of each individual droplet/particle to move freely through the aqueous phase are primarily responsible for the observed RTD. In other words, in the absence of no-slip conditions there would be little or no velocity profile. However, diffusive effects could still impart a residence time distribution and that is the governing factor here. One other point should be highlighted for reasons that will become apparent in the forthcoming discussion of the polymerizations. When viewed on a physical length scale, the dispersion here is quite broad. The first traces of dye begin to show up at roughly τ - 0.37τ and level off around τ + 0.37τ. In terms of length the range of the dispersion is 2 × 0.37 L_r, or ~ 4500 cm.

One laboratory technique that is used to induce ideal flow without the high flow rates normally required is the introduction of an inert gas, for example nitrogen, into the fluid flow.[37,38] If metered properly, the fluid will separate into individual elements, small cylinders or "plugs" (See Figure 4.6). Provided that the plugs are short compared to the length of the reactor, axial dispersion is minimized and, in principle at least, flow profiles approaching ideal flow can be achieved. This regime is referred to in this work as isolated plug flow so as to avoid confusion with what is classically known as plug flow. In order to insure the assumption of ideal flow, a tracer experiment was performed on these isolated plugs. The results are shown in Figure 4.7 and compared to both ideal and laminar flow. While the data reveal some deviation, on the whole they demonstrate that for this system the assumption of ideality holds.

Figure 4.6 Miniemulsion using metered nitrogen to induce isolated plug flow. In this instance, the length of the "plugs" moving through the reactor tubing is ~ 1 cm.

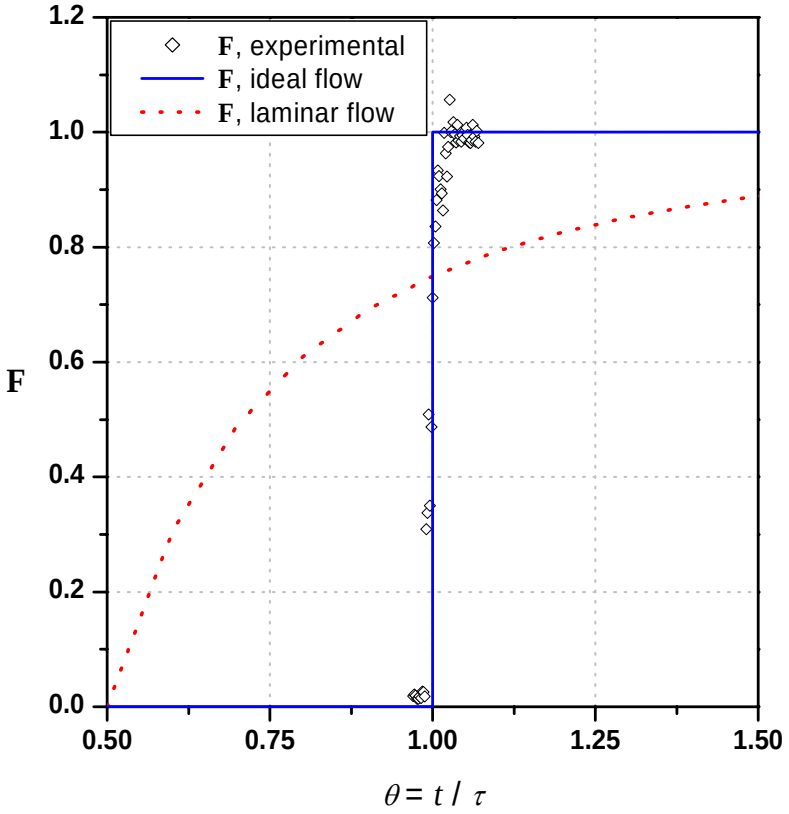


Figure 4.7 Experimental and theoretical step response curves of miniemulsion in isolated plug flow.

4.2.3 Polymerizations

In order to assess the influence of the flow regime on the PDI of the polymer, polymerizations of styrene were conducted in the two regimes previously mentioned. The results are summarized in Table 4.2. As expected with non-ideal flow, the PDI of the polymer formed is higher than that synthesized in batch. In each of runs T1-T4 the PDI is significantly higher (5.4-7.7%) than the corresponding batch runs, B1-B4. Provided all else is equal, this would be attributable to the residence time distribution imparted to the

droplet/particles as they flow through the reactor. In a conventional free radical polymerization, the lifetime of a growing chain is relatively short, on the order of 1 s. As

Table 4.2 Comparison of Batch and Tube Number Average Molecular Weights and PDIs.

Run	Flow Regime	Conversion	M_n, theo	M_n, actual	M_w/M_n	Δ%
T1	Simple	57%	30388	28056	1.37	5.4
B1	—	57%	30468	30299	1.30	
T2	Simple	58%	30621	28839	1.38	7.0
B2	—	58%	30590	29731	1.29	
T3	Simple	91%	48283	45056	1.35	7.1
B3	—	92%	48774	48068	1.26	
T4	Simple	92%	48921	46398	1.40	7.7
B4	—	93%	49002	48435	1.30	
T5	Isolated plug	22%	11526	11846	1.38	-2.1
B5	—	24%	10389	12786	1.41	
T6	Isolated plug	52%	26251	26800	1.66	0.6
B6	—	52%	26694	27725	1.65	
T7	Isolated plug	89%	45433	46081	1.26	4.1
B7	—	91%	46359	44995	1.21	

such, residence time distributions have little effect on PDIs In a controlled polymerization, however, the lifetime of a growing chain can approach its residence time in the reactor. To achieve a narrow PDI, the chains must all initiate quickly and grow at the same rate for the same amount of time. With a residence time distribution, this cannot occur. The degree of the flow non-ideality will dictate the degree to which the PDI is affected. It follows that in the limit of ideal flow, where there is no residence time distribution, the PDI of the polymer will be identical to that formed in a batch reactor. Runs T5-T7 and B5-B7 compare polymer formed in batch to polymer formed in the tubular reactor via segregated plugs. The PDIs of the polymers formed are much closer,

and in the case of T5/B5, the PDI of the polymer formed in the tube was slightly lower than that formed in the batch reactor.

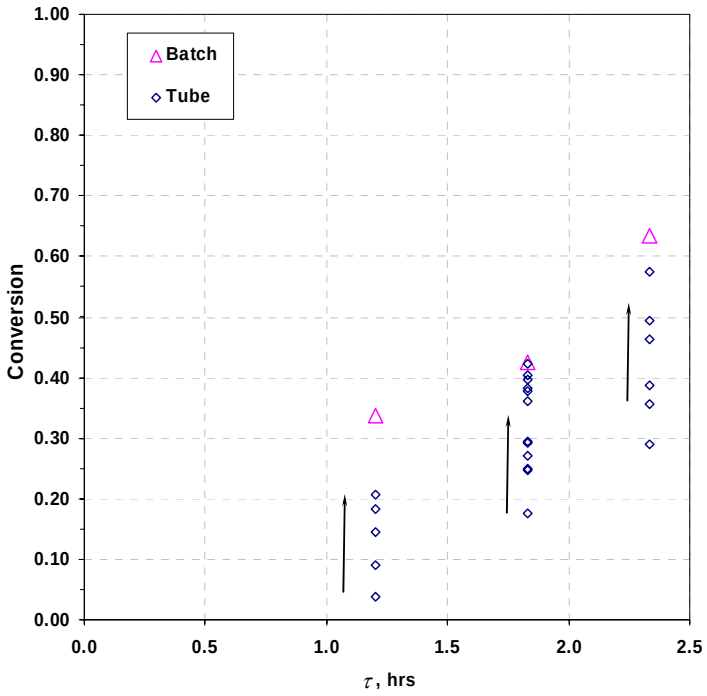


Figure 4.8 Conversion vs. average residence time, tube and batch samples. Tube reactor in isolated plug flow (~ 1 cm plug miniemulsion, ~ 2 cm N_2). Arrows indicate tube samples taken as time progressed, conversion in the tube starts lower than batch and gradually increases.

One interesting finding that was unexpected was the transient behavior of the tube reactor while in isolated plug flow. Given a stream of equally spaced and completely isolated small plugs of reactant flowing through the reactor, one expects that if they all have the same residence time that each plug will exit the reactor with the same

conversion. Each plug would behave as a batch reactor and its average residence time in the reactor would be equivalent to time in a batch reactor. In other words, there would be no transient behavior because there is no steady state to be reached. In this case, something different was observed in practice. Figure 4.8 shows the results of three different polymerizations conducted at three different residence times. In each case the miniemulsion was in isolated plug flow using metered N_2. The arrows indicate the general trend in conversion as the plugs exited the reactor. The first plugs out were always at much lower conversions than batch and the conversion increased as a function of time (elapsed time, t, not residence time, τ). In order to assess this behavior, more complete sets of data were collected and are shown in Figure 4.9. In these two experiments, samples were taken starting at $t + \tau$ at time intervals of $\Delta t \sim \tau/100$ until the charge in the plug generator was exhausted ($\sim \tau/10$). The dotted lines represent the ideal, or batch, conversion for the residence time employed. Those ideal conversions were determined from concurrent batch experiments and are presented as a reference point. What they show is a reactor exhibiting transient start-up behavior, where conversions start out much lower than predicted, increase over time and eventually reach a steady state at batch conversions. It was thought initially that residual dye in the tubing may have some affect on the reaction. However, after replacing all of the tubing, the behavior persisted. A common practice that might otherwise account for such behavior is to start the reactor with the tubes full of water, but in this case extreme care was exercised in order to ensure that every component of the system (tube, pumps, fittings, etc.) was completely clean and dry before starting. Additionally, the entire system was purged with N_2 for several hours beforehand to prevent any effect on the polymerization from oxygen

contamination. Assuming that the system is neither diluted nor contaminated, then two observations can be made about the data. First, in view of the nature of the system one might imagine that given the surface to volume ratio of the small plugs

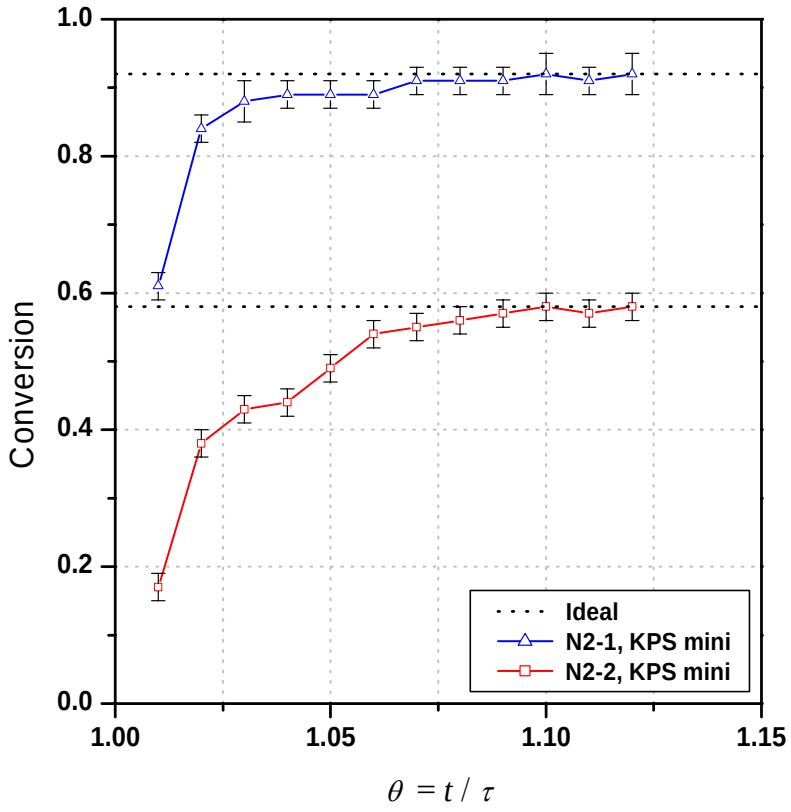


Figure 4.9 Conversion profile in tube reactor using miniemulsion in isolated plug flow (~ 1 cm plug miniemulsion, ~ 2 cm N_2) Last two data points in each series had no N_2, i.e., no plugs.

(~ 2 cm^{-1} for a 1 cm plug) some surface effect might be occurring that contributes to slower kinetics. However, if that alone was the cause then the conversion would be expected to remain at depressed levels. Each plug would exhibit similar kinetics regardless of when it entered and exited the reactor. In other words, in the complete absence of other effects, if something were happening independently in each individual plug then the conversion profile should be flat, but at a lower level than batch. This points towards the second hypothesis, that given the nature of the transient behavior, the plugs must be interacting in some manner.

In light of this supposition, an observation made during the polymerizations and initially thought to be inconsequential becomes important. It was noted that occasionally very small droplets were left behind by the plugs as they moved through the reactor. Since the droplets appeared transparent, they would likely be composed primarily of water and dissolved KPS, containing only relatively small numbers of particles. If that were the case then it is possible that the miniemulsion plugs were being "stripped" of water and initiator as they traversed the length of the tube. If significant, it might explain the transient behavior. As later plugs pick up the stripped initiator left behind by the earlier plugs, the effect would be the most pronounced on the first plugs through. The system would come to some steady state level after a sufficient number of plugs traversed the reactor.

To test the basic hypothesis, two experiments were conducted at different average residence times using solution polymerization instead of miniemulsion. Since the solution

system is homogeneous, the small droplets left behind would have the same composition as the larger plugs. However because the droplets remain stationary while the plugs continue to move, their residence time would be slightly longer than the plugs they would encounter. This would tend to increase the overall conversion as compared to batch at the same average residence time. In this case, the effect would be different from the miniemulsion case. The first plugs out should be at the same conversion as batch because they have had few or no droplet encounters. The conversion would increase from this point to some steady state level. The results, which are shown in Figure 4.10, give support to the droplet theory. Again, the samples were taken starting at $t + \tau$ at time

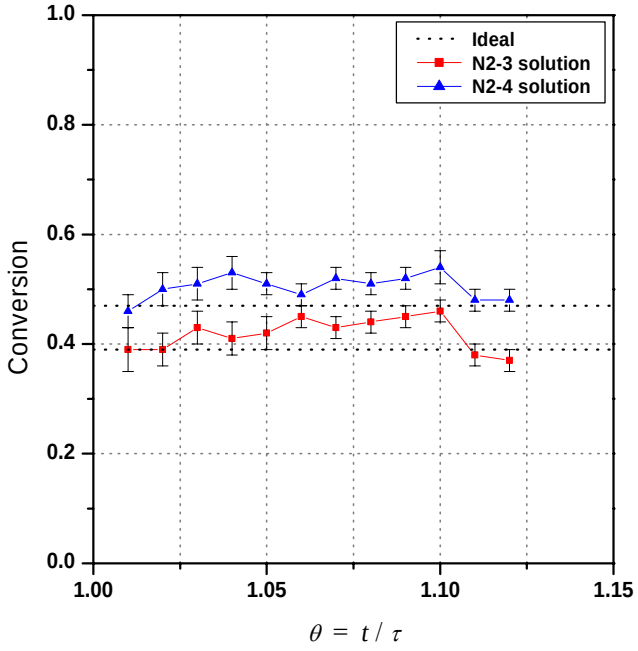
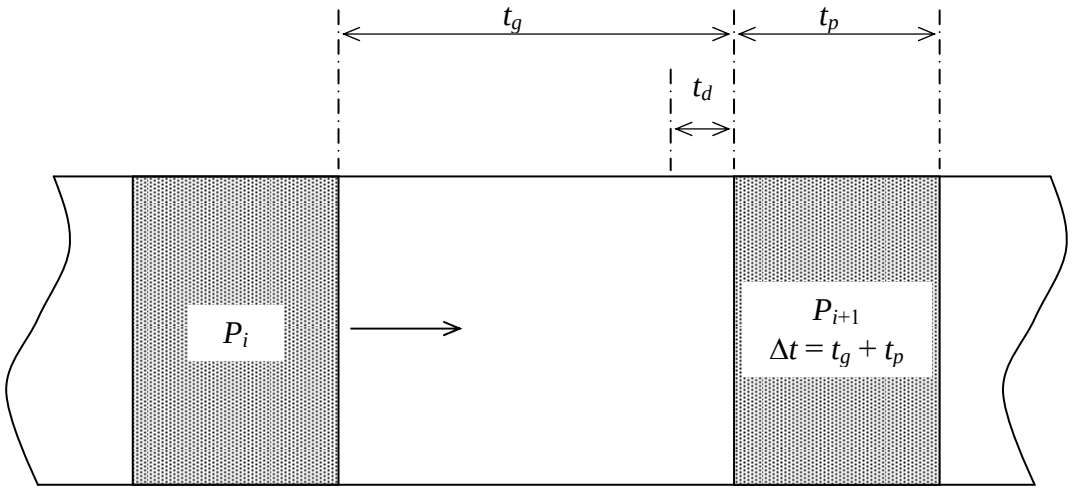
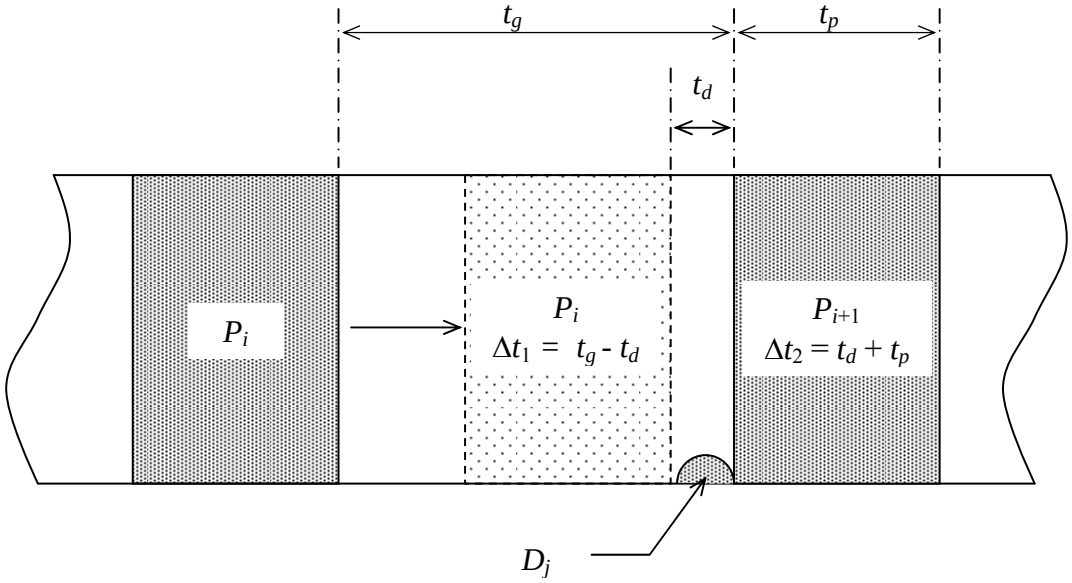


Figure 4.10 Conversion profile in tube reactor using solution polymerization in isolated plug flow (~ 1 cm plug solution, ~ 2 cm N_2) Last two data points in each series had no N_2, i.e., no plugs.



(a)



(b)

Scheme 4.1 Model development diagram of plug interaction in a tubular reactor. (a) No droplet encounter (b) Droplet encounter

intervals of $\Delta t \sim \tau/100$ until the charge in the plug generator was exhausted ($\sim \tau/10$). The conversion of the first plugs out is almost identical to batch at the same average residence time and the conversion increases slowly from then on as time passes. The last two data points before the end of each run were from samples taken in the normal flow regime, i.e. without the use of metered N_2. In other words, the nitrogen flow was stopped, eliminating the formation of plugs. The data show that the conversion quickly drops to that predicted by the average residence time of the solution.

In an effort to further scrutinize the theory, a model of the initiator concentration and monomer conversion at the outlet of the reactor was developed. Scheme 4.1 shows the form and basic notations for model development. If no droplets are present, then the system is represented by Scheme 4.1(a) where $\Delta t = t_g + t_p$ and:

$$\frac{dI_i^P}{dt} = -k_d I_i^P \qquad (4.7)$$

The defining event, Scheme 4.1(b), occurs after plug P_i travels $\Delta t = t_g - t_d$ across the gap between it and the preceding plug, P_{i+1}, at which point it encounters droplet, D_j. During $\Delta t = t_d + t_p$, P_i incorporates and releases D_j. For all plugs except the first plug through the reactor this event can be represented mathematically by:

$$V_i^P \frac{dI_i^P}{dt} = q_D I_j^D - q_D I_i^P - V_i^P k_d I_i^P \qquad (4.8)$$

One simplification is the assumption that the plug volume remains constant at V_0 for all but the first plug through the reactor. For the first plug through the reactor, recognizing that the volume would change with each droplet it releases:

$$V_1^P \frac{dI_1^P}{dt} = -q_D I_1^P - V_1^P k_d I_1^P \qquad (4.9a)$$

where

$$V_1^P = V_0 - j V_D \qquad (4.9b)$$

Finally, for the droplets during the time $\Delta t = t_g - t_d$:

$$\frac{dI_j^D}{dt} = -k_d I_j^D \qquad (4.10)$$

All conversion occurs in the plugs and is represented by:

$$\frac{dX_i}{dt} = k_p (1 - X_i) \left(\frac{2 f k_d I_i^P}{k_t} \right)^{\frac{1}{2}} \qquad (4.11)$$

This assumes conventional free radical first order kinetics where chain transfer is not taken into account. For P_1, when there is no droplet release (4.7) and (4.11) are integrated across $\Delta t = t_g + t_p$. When P_1 releases, (4.7) and (4.11) are first integrated across $\Delta t = t_g - t_d$, followed by integrating (4.9a) and (4.11) across $\Delta t = t_d + t_p$. The concentration of D_j is set at I_1^P and the volume is incremented down via (4.9b.) and substituted back into (4.9a) during the next P_1 droplet release. For plugs P_2 to P_n a similar calculation is carried out

using (4.7), (4.8), (4.10) and (4.11) noting that now the plug volume remains constant at V_0. If there is no droplet encounter, the plug is integrated across $\Delta t = t_g + t_p$ and incremented to I_{i+1}^P and X_{i+1}, respectively. For droplet encounters, the plug and droplet are both integrated across $\Delta t = t_g - t_d$ using (4.7), (4.10) and (4.11). Next, they are integrated using (4.8) and (4.11) across $\Delta t = t_d + t_p$ after which the plug is incremented to I_{i+1}^P and X_{i+1} and I_j^D is set to I_{i+1}^P. For each plug that traverses the reactor the process is repeated $\tau / (t_g + t_p)$ times. The parameters used in the model are shown in Table 4.3 and reflect the actual lengths, tube inner diameter and average velocity employed in one of the original experiments. In constructing the model, the droplet volumes are also assumed constant

Table 4.3 Isolated plug model parameters

Quantity	Value	Units	Ref
L_r	6000	cm	
L_p	1	cm	
L_g	2	cm	
u	0.4	$cm \cdot s^{-1}$	
k_d	2.2×10^{-5}	s^{-1}	[39]
k_p	475	$L \cdot mol^{-1} \cdot s^{-1}$	[33]
k_t	6.1×10^7	$L \cdot mol^{-1} \cdot s^{-1}$	[34]
d_i	0.1651	cm	
V_d	$V_{p,1}/100$	cm^3	
$V_{p,1}$	$\pi/4 \cdot d_i^2 \cdot 1$ cm	cm^3	
n_p/n_d	25		
f	0.65		

and of the same volume and size regardless of the volume of the plug that released it. The droplet volume, V_d, was estimated based upon observation of the actual droplets and was held constant at $V_{p,1}$ /100 where $V_{p,1}$ is the volume of a plug 1 cm in length. They are modeled to be a hemisphere so that t_d is constant and simply a function of the diameter of the sphere that would be formed from two droplets. The model also assumes that the droplets are formed only by releases from the first plug through and that no new droplets are created (or existing droplets disappear) afterward. This assumption is based upon observing the behavior of the actual droplets in the reactor. The term n_p/n_d is the ratio of the total number of plugs to the total number of droplets and was also estimated based upon observation.

Figure 4.11 shows the results produced by the model at 4 different plugs lengths. The results show that both the initiator concentration and the overall conversion are adversely affected as the length of the plug decreases, behavior similar to that observed experimentally. Increasing the plug length, and therefore the plug volume, very quickly reduces the transient behavior. At plug lengths of 10 cm, the behavior is almost eliminated. A similar trend occurs by increasing the diameter of the tubing, as shown in Figure 4.12. This may explain in part why this phenomenon has not been reported in past studies using isolated plug flow. In the work of Poehlein et al.,[38,40] the tubing size was at least twice that used in this study. It also has important implications for future work that might include utilizing even smaller tube diameters.

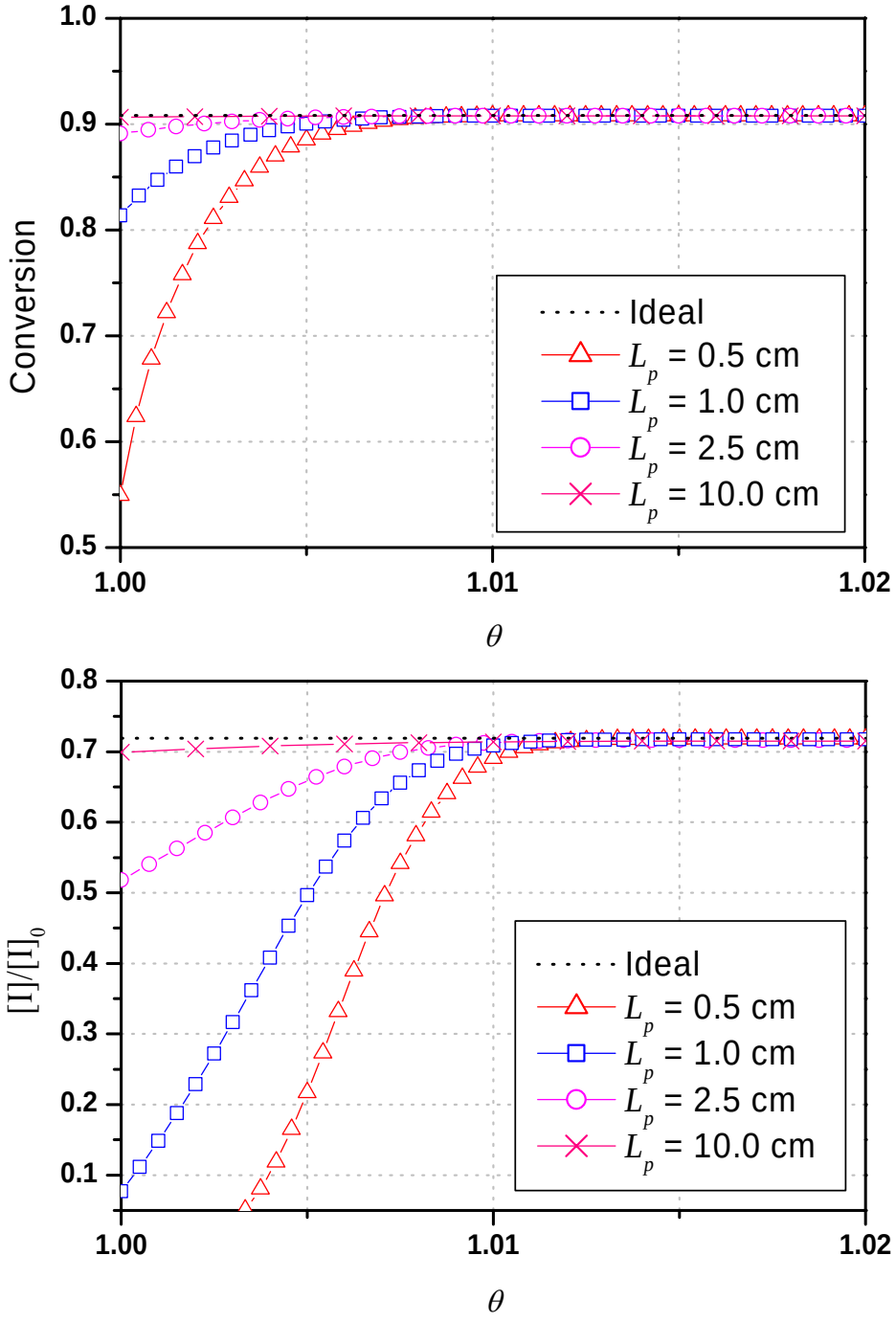


Figure 4.11 Evolution of outlet initiator concentration and conversion with dimensionless residence time in isolated plug flow in a tubular reactor. Results shown at 4 different plug lengths.

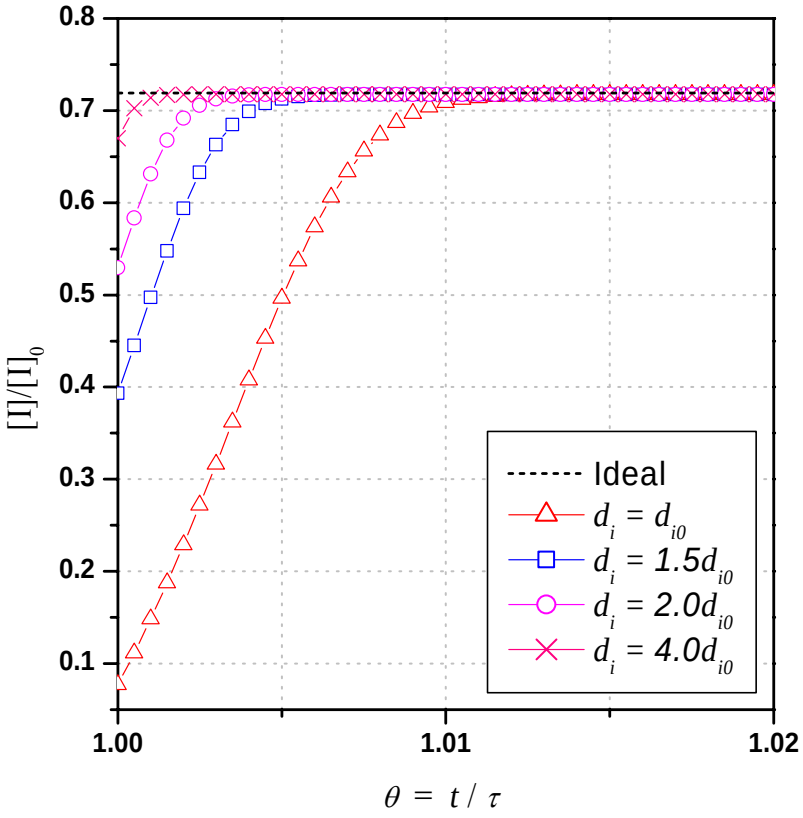


Figure 4.12 Effect of tube diameter on the outlet initiator concentration in isolated plug flow in a tubular reactor. Results shown at 4 different tube diameters.

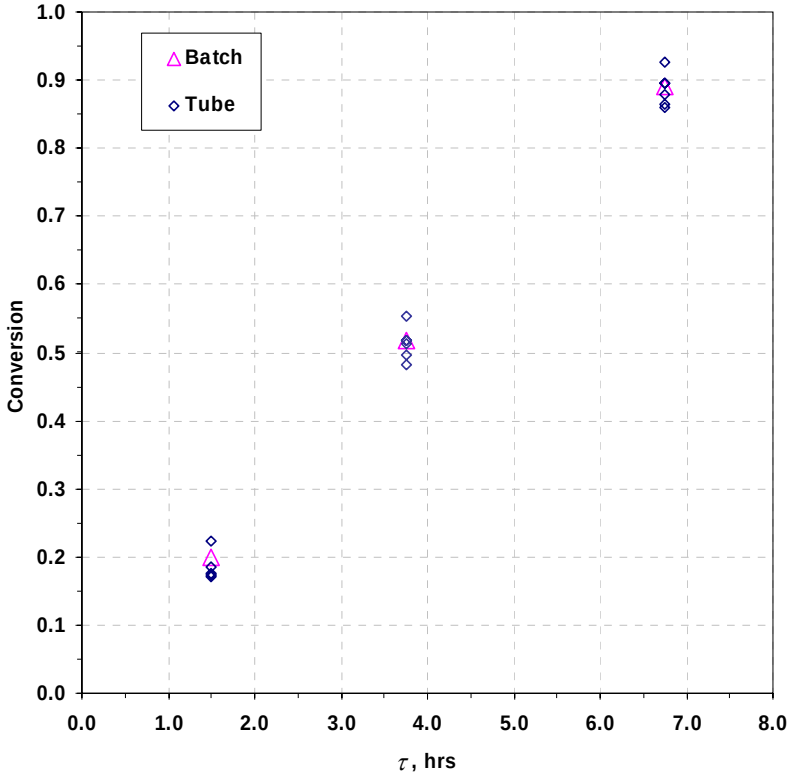


Figure 4.13 Conversion vs. average residence time, tube and batch reactions. Tube samples taken over time, starting with first plugs out. Tube and batch conversions are similar with no transient conversion drift. Plug size ~ 20 cm.

Using these insights, by increasing the length of the plugs in the actual reactor the transient behavior was essentially eliminated. The results of three different polymerizations are shown in Figure 4.13. In this case, the plug length was increased to ~ 20 cm. The data show that the conversion in the tube is similar to that in batch from $t = \tau$ onward. While some scatter exists, there is no drift in the conversion as is seen in Figure 4.8. The first plugs out are at similar conversions to batch and the conversion remains at that level as the following plugs exit the reactor.

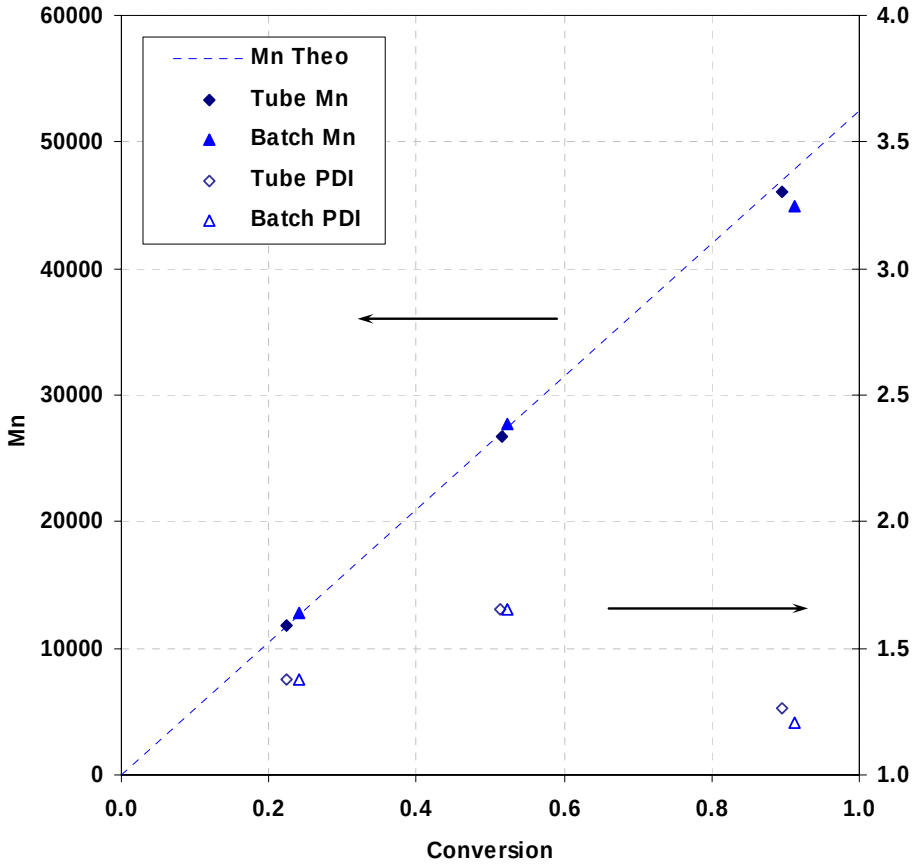


Figure 4.14 Number average molecular weight and PDI as a function of conversion. Three separate experiments comparing batch to tube using metered N_2 with ~ 20 cm plugs.

4.3 Conclusions

In this chapter, the results of studies analyzing the flow characteristics of a tubular reactor and their effects upon polymer formed using RAFT/miniemulsion were presented. A bulk RAFT system was modeled in laminar flow and the results compared to a batch system. It was shown that although a significant difference in polymer properties between batch and tube can occur when the tube is in laminar flow, the variance in the actual experimental results were not entirely consistent with the model, prompting further investigation. The actual flow profile in the tubular reactor was then determined using a modified dye tracer approach. It was shown that by utilizing an oil soluble dye, in conjunction with SEC/UV, the droplet/particle flow behavior of a miniemulsion could be quantified. Dye tests performed in normal flow at low Reynolds numbers revealed that the reactor was not operating in laminar flow but that the axial dispersion was quite high. The vessel dispersion number was calculated to be ~ 0.1. Dye tests carried out in isolated plug flow demonstrated the near-ideal nature of the flow regime. Polymer formed by RAFT polymerizations in this regime was shown to have similar PDIs as that formed in batch. Taken together this establishes a direct relationship between the residence time distribution and final polydispersity of the polymer formed in a controlled radical polymerization. Finally, transient behavior in isolated plug flow was demonstrated to be caused by loss of water and initiator by the plugs as they traversed the reactor. It was further shown that the effects could be minimized by either increasing the plug length or the tube diameter.

4.4 Experimental

4.4.1 Materials

Styrene (monomer, ≥ 99.0%, Aldrich) was cleaned by either vacuum distillation or by passing through a column packed with inhibitor remover. The column packing was purchased from Aldrich and was specific to the type of inhibitor in the monomer. Potassium persulfate (initiator, KPS, ≥99.0 %, Aldrich), Triton X-405 (non-ionic surfactant, TX405, 70% solution in water, Aldrich), hexadecane (co-stabilizer, ≥ 99.0%, Aldrich), and sodium dodecyl sulfate (ionic surfactant, SDS, ≥99%, Aldrich) were used as received. The RAFT agent, 1-phenylethyl phenyldithioacetate (PEPDTA) was synthesized according to the procedure outlined in Chapter 3, Section 3.4.2. The reagents for the RAFT agent synthesis, hydrochloric acid (37% in H_2O, Aldrich), magnesium sulfate (≥ 99%, Aldrich), *p*-toluenesulfonic acid (≥ 98%, Aldrich), benzyl chloride (≥ 99.9%, J.T. Baker), carbon disulfide (≥ 99.9%, J.T. Baker), carbon tetrachloride (≥ 99.9%, Aldrich), anhydrous diethyl ether (≥ 99.9%, Fisher) ,magnesium turnings (≥ 98.0%, Aldrich) and tert-butyl catechol (≥ 99%, Aldrich) were used as received. The oil-soluble dye, *N*-ethyl-*N*-(2-hydroxyethyl)-4-(4-nitrophenylazo)aniline (Disperse Red 1, ≥ 99.9%, Aldrich), was also used as received. Deionized water was generated in-house with a U.S. Filter Systems Deionizer and was used without further purification.

4.4.2 Miniemulsion preparation

The recipe for the miniemulsion polymerizations is shown in Table 4.4. The surfactants, SDS and TX405, were first added to water and allowed to mix for 15 minutes. The monomer, hexadecane, and RAFT agent were combined and allowed to mix for 15 minutes. The organic phase was then added to the aqueous phase and agitated

vigorously with a magnetic stirrer for 30 minutes, forming a light yellow emulsion. The miniemulsion was formed by sonicating for 20 minutes (Fisher 300 Sonic Dismembrator at 70% output). During the sonication, the miniemulsion was cooled by an ice bath in order to keep the temperature low so that any thermal initiation of the monomer would be minimized.

Table 4.4 Miniemulsion recipe

Component	Mass, g	Basis
Water	80 g	
Styrene	20 g	25 wt-% of water
Triton X-405	0.1 g	0.005 mol/L (based on aqueous phase)
SDS	0.9 g	"
Hexadecane	0.4 g	2 wt-% of monomer
PEPDTA	0.1 g	$[Sty]_0/[PEPDTA]_0 = 500$
KPS	0.10 g	$[PEPDTA]_0/[KPS]_0 = 1$
Temperature	70 ºC	

4.4.3 Dye saturated miniemulsion preparation

The recipe for the dye saturated miniemulsion is shown in Table 4.5. The styrene was further stabilized by the addition of tert-butyl catechol to 0.5 wt-%. The dye was then dissolved in the styrene, followed by the addition of hexadecane. After mixing the surfactants in water, the aqueous and organic phases were sonicated (20 minutes, 70% output) together while agitated by a magnetic stirrer. During sonication the miniemulsion was cooled via ice bath.

Table 4.5 Recipe for dye-saturated miniemulsion

Component	Mass, g	Basis
Water	50.0	
Styrene	50.0	100 wt-% of water
Triton X-405	0.58	0.005 mol/L (in aqueous phase)
SDS	0.07	"
Hexadecane	1.0	2 wt-% of monomer
Disperse Red 1	1.0	2 wt-% of monomer

4.4.4 Modified dye tracer experiments, isolated plug flow

A general schematic illustration of the set up for conducting the modified tracer experiments is shown in Figure 4.15. The miniemulsion was fed into the reactor via an FMI QG 50 laboratory pump outfitted with a Kynar® pump head with carbon liner and ⅛-inch SS piston. The pump was calibrated beforehand using deionized water and by measuring the time to fill a 5 mL volumetric flask at four different pump settings. A syringe pump was utilized to inject the dye saturated miniemulsion into the reactor. All the tubing used was 1/8" OD - 1/16" ID PFA (perfluoroalkoxy, a copolymer of TFE). The reactor tubing was arrayed in helical coils (~25 cm in diameter) and submerged horizontally in a constant temperature, 75 liter water bath. For all the dye experiments the bath was at room temperature. During the isolated plug flow experiments, a coil used for plug generation (referred to as the plug generator) was included just before entering the reactor. This was incorporated because of the difficulty of maintaining a constant volume flow of nitrogen on a continuous basis. The length of the plug generator was 1/10 the length of the reactor. The feed was first started then the nitrogen added to begin

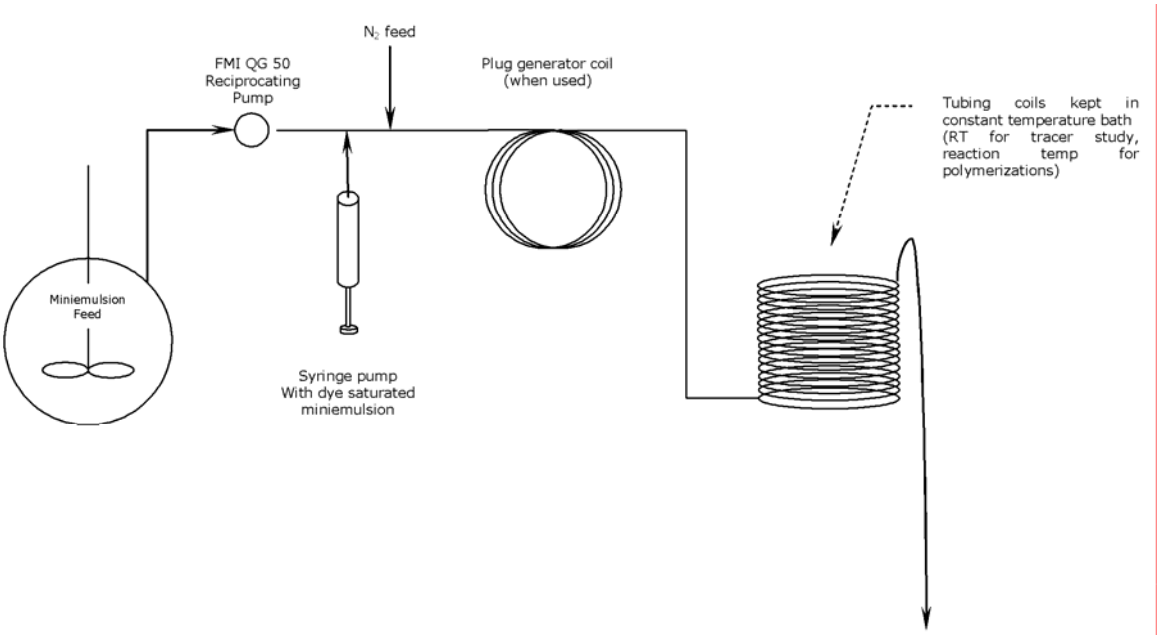


Figure 4.15 General schematic illustration of the tubular reactor utilized in this study. Set up was used for tracer studies and polymerizations.

forming the isolated plugs. The nitrogen flow was metered down by use of a needle valve located after the regulator on the nitrogen tank. After filling the plug generator to ~ ½ full, the dye saturated miniemulsion was injected by starting the syringe pump. When the generator was completely filled with plugs, the feed and syringe pumps were stopped. The feed pump was started again after all movement in the generator ceased. This allowed the determination of t_0 and τ by simply marking the reactor entrance and exit times of the first plug in the series. Samples were taken pre-determined intervals, noting that in this case since the length of the plug generator was $L_r/10$, the time frame for sampling was $\tau \pm \tau/20$.

4.4.5 Modified dye tracer experiments, simple tubular flow

In this instance, the plug generator coil was bypassed and no nitrogen was metered into the reactor, creating a constant, unbroken flow of miniemulsion. The dye-containing miniemulsion was injected at a known time and samples were subsequently taken at predetermined intervals. Since the Reynolds number was low enough to suggest laminar flow, ~ 10-20, the range of sampling was $\tau \pm \tau/2$.

4.4.6 Polymerizations in the tubular reactor

Polymerizations were conducted in a similar manner as the dye experiments, using the configuration shown in Figure 4.15 without the syringe pump. The reaction system was altered as mentioned in the prior section depending upon whether or not isolated plug flow was desired. After preparing the miniemulsion, it was placed in the feed tank and the requisite amount of initiator was added to the vessel. In order to suppress initiation before the feed entered the reactor, the feed vessel was kept at ~2 ºC via submersion in a refrigerated water bath. Refrigeration and circulation were supplied by a VWR 1186D 28L programmable heating/cooling circulating bath. The feed was kept agitated via magnetic stirring and under ultra high purity nitrogen throughout the experiment. For comparison to batch, feed samples were sealed in vials, purged with nitrogen, placed in the reactor water bath and polymerized for the requisite time.

The water bath was heated by a submersible heating coil and the temperature controlled with a thermocouple and temperature controller. Because of the high heat transfer inherent in tube reactors and the small size of the tubing in relation to the water bath, the temperature was assumed to be constant throughout the length of the submerged tubing. In order to insure that the inside of the tubing was completely dry before running

the reaction, the reactor was filled with THF at room temperature, pumped dry, heated to reaction temperature and purged with nitrogen overnight. The transparency of the tubing facilitated inspection for plugging and fouling.

4.4.7 Characterization

The samples from the tracer experiments were dried for 24 hours in a vacuum oven (50 ºC, ~ 100 kPa vacuum) and the residue was dissolved in 1 mL of THF. They were then analyzed using SEC/UV with the UV detector set at 503 nm. This is represents the absorption max of the dye and does not overlap with other system components. In this manner only the dye is seen in the UV. The areas under the UV peaks were determined and normalized for concentration using the area under the refractive index (proportional to total mass) peak of the polymeric surfactant, TX405.

After quenching with hydroquinone, polymer latex samples were dried in a consistent manner and monomer conversion was subsequently determined gravimetrically. Samples were dissolved in THF and run through a pipette column packed with alumina to remove the TX405 polymeric surfactant. The number average molecular weight, M_n, and the polydispersity (PDI), M_w/M_n, were calculated using data gathered via size exclusion chromatography (SEC-Viscometry-RALLS) with THF as eluent. Three columns (American Polymer Standards styrene-divinylbenzene 100 Å, 1000 Å, and 10^5 Å) mounted in a Waters WAT038040 column heater set at 30 ºC were utilized. The columns were connected to a Viscotek GPCMax pump/autoinjector, a Viscotek T60A dual detector (viscometer and light scattering), a Waters 410 refractive index detector, an LDC Milton Roy Spectromonitor 3000 UV detector (at 254, 311, or

503 nm). Latex particle sizes and polydispersities were analyzed using quasi-elastic light scattering (QELS, Protein Solutions DynaPro99 with DynaPro DCS v 5.26 software).

4.5 References

1. Russum, J. P.; Jones, C. W.; Schork, F. J. *Macromol. Rapid Comm.* **2004**, *25*, 1064.

2. Russum, J. P.; Jones, C. W.; Schork, F. J. *Ind. Eng. Chem. Res.* **2005**, *44*, 2484.

3. Barner-Kowollik, C.; Quinn, J. F.; Morsley, D. R.; Davis, T. P. *J. Polym. Sci. Polym. Chem.* **2001**, *39*, 1353.

4. Drache, M.; Drees, K.; Schmidt-Naake, G. *DECHEMA Monographien* **2004**, *137*, 479.

5. Feldermann, A.; Toy, A. A.; Davis, T. P.; Stenzel, M. H.; Bamer-Kowollik, C. *Polymer* **2005**, *46*, 8448.

6. Monteiro, M. J. *J. Polym. Sci. Pol. Chem.* **2005**, *43*, 3189.

7. Chaffey-Millar, H.; Busch, M.; Davis, T. P.; Stenzel, M. H.; Barner-Kowollik, C. *Macromol. Theor. Simul.* **2005**, *14*, 143.

8. Coote, M. L.; Radom, L. *J. Am. Chem. Soc.* **2003**, *125*, 1490.

9. Coote, M. L. *J. Phys. Chem. A* **2005**, *109*, 1230.

10. Kwak, Y.; Goto, A.; Komatsu, K.; Sugiura, Y.; Fukuda, T. *Macromolecules* **2004**, *37*, 4434.

11. Wulkow, M.; Busch, M.; Davis, T. P.; Barner-Kowollik, C. *J. Polym. Sci. Pol. Chem.* **2004**, *42*, 1441.

12. Wang, A. R.; Zhu, S. P. *Macromol. Theor. Simul.* **2003**, *12*, 663.

13. Wang, A. R.; Zhu, S. P. *Macromol. Theor. Simul.* **2003**, *12*, 196.

14. Tonge, M. P.; McLeary, J. B.; Vosloo, J. J.; Sanderson, R. D. *Macromol. Symp.* **2003**, *193*, 289.

15. Perrier, S.; Barner-Kowollik, C.; Quinn, J. F.; Vana, P.; Davis, T. P. *Macromolecules* **2002**, *35*, 8300.

16. Vana, P.; Davis, T. P.; Barner-Kowollik, C. *Macromol. Theor. Simul.* **2002**, *11*, 823.

17. Barner-Kowollik, C.; Quinn, J. F.; Nguyen, T. L. U.; Heuts, J. P. A.; Davis, T. P. *Macromolecules* **2001**, *34*, 7849.

18. Zhang, M.; Ray, W. H. *Ind. Eng. Chem. Res.* **2001**, *40*, 4336.

19. Wang, A. R.; Zhu, S. P. *J. Polym. Sci. Polym. Chem.* **2003**, *41*, 1553.

20. Barner-Kowollik, C.; Coote, M. L.; Davis, T. P.; Radom, L.; Vana, P. *J. Polym. Sci. Pol. Chem.* **2003**, *41*, 2828.

21. Wang, A. R.; Zhu, S. P.; Kwak, Y. W.; Goto, A.; Fukuda, T.; Monteiro, M. S. *J. Polym. Sci. Pol. Chem.* **2003**, *41*, 2833.

22. Barner-Kowollik, C.; Davis, T. P.; Heuts, J. P. A.; Stenzel, M. H.; Vana, P.; Whittaker, M. *J. Polym. Sci. Pol. Chem.* **2003**, *41*, 365.

23. Monteiro, M. J.; Bussels, R.; Beuermann, S.; Buback, M. *Aust. J. Chem.* **2002**, *55*, 433.

24. Vana, P.; Quinn, J. F.; Davis, T. P.; Barner-Kowollik, C. *Aust. J. Chem.* **2002**, *55*, 425.

25. Kwak, Y.; Goto, A.; Tsujii, Y.; Murata, Y.; Komatsu, K.; Fukuda, T. *Macromolecules* **2002**, *35*, 3026.

26. Monteiro, M. J.; de Brouwer, H. *Macromolecules* **2001**, *34*, 349.

27. Levenspiel, O., *Chemical Reaction Engineering*. 3rd ed.; Wiley: New York, NY, 1999.

28. Nauman, E. B., *Chemical Reactor Design, Optimization, and Scaleup*. McGraw-Hill: New York, 2002.

29. Kim, D. M.; Nauman, E. B. *Ind. Eng. Chem. Res.* **1997**, *36*, 1088.

30. Zhang, M.; Ray, W. H. *J. Appl. Polym. Sci.* **2002**, *86*, 1047.

31. Lansalot, M.; Davis, T. P.; Heuts, J. P. A. *Macromolecules* **2002**, *35*, 7582.

32. Brandrup, J.; Immergut, E. H., *Polymer Handbook*. 3rd ed.; Wiley: 1989.

33. Buback, M.; Gilbert, R. G.; Hutchinson, R. A.; Klumperman, B.; Kuchta, F. D.; Manders, B. G.; Odriscoll, K. F.; Russell, G. T.; Schweer, J. *Macromol. Chem. Physic.* **1995**, *196*, 3267.

34. Mahabadi, H. K.; Odriscoll, K. F. *J. Macromol. Sci. Chem.* **1977**, *A11*, 967.

35. Hutchinson, M.; De Hoog, F. *Numer. Math.* **1985**, *47*, 99.

36. van der Laan, E. T. *Chem. Eng. Sci.* **1958**, *7*, 187.

37. Poehlein, G. W.; Schork, J. *Trends in Polymer Science (Cambridge, United Kingdom)* **1993**, *1*, 298.

38. Shoaf, G. L.; Poehlein, G. W. *Polym.-Plast. Technol.* **1989**, *28*, 289.

39. Gilbert, R. G., *Emulsion Polymerization : A Mechanistic Approach*. Academic Press: 1995.

40. Lee, H. C.; Poehlein, G. W. *Chem. Eng. Sci.* **1986**, *41*, 1023.

CHAPTER 5

SUMMARY AND SUGGESTIONS FOR FURTHER INQUIRY[†]

5.1 Summary

Controlled radical polymerization finally provides a means to produce polymers with complex, designed architectures with the robustness and tolerance of conventional free radical chemistry. This allows for a much broader range of monomers and functionalities than are available via historically utilized anionic techniques and more importantly for conducting the polymerizations in solvent-free aqueous dispersed systems. Of the currently prevalent CRPs, Reversible Addition Fragmentation Chain Transfer (RAFT) has been shown to be extremely versatile in terms of range of monomers and their functionalities, along with the ability to be easily employed in mild reaction conditions. In an effort to gain further understanding in employing RAFT in aqueous dispersed systems, the work presented has attempted to expand upon the established foundation in two directions previously unexplored. First, that of vinyl acetate homopolymerization in a RAFT/miniemulsion system and second in advancing from batch systems to continuous reaction systems, specifically tubular reactors.

In terms of the cumulative research output to date regarding RAFT polymerization, vinyl acetate occupies only a small percentage. The primary reason has been the lack of good RAFT agents which could successfully polymerized the highly reactive vinyl acetate radical. Recent studies utilizing xanthates and dithiocarbamates have shown

[†] Portions of this chapter have been previously published, *J. Polym. Sci. Polym. Chem.* **2005**, *43*, 2188.

promise, however understandably they were conducted exclusively in either bulk or with some organic solvent. The investigation here takes the next logical step by combining the RAFT polymerization of vinyl acetate with miniemulsion, an aqueous dispersed system well suited to CRPs. Using an oil-soluble initiator to suppress aqueous phase nucleation, it was shown that using two different xanthates, vinyl acetate could be polymerized in a controlled manner in miniemulsion, albeit with complications. Primarily, much slower kinetics and limiting conversions were observed in miniemulsion as compared to the same recipe in bulk (which achieved ~ 100% conversion). Molecular weights exhibited the characteristic linear growth with conversion, but the polydispersities were higher in miniemulsion than bulk. Gel formation and glass effects were ruled out as a source for the limiting conversions, as were hydrolysis/decomposition of the RAFT agent and desorption of several of the smaller molecular species that are formed early in the polymerization. The use of a water soluble initiator was demonstrated to overcome most of the difference observed, however limiting conversions at similar initiator levels to bulk remained an issue.

To demonstrate viability in continuous systems, RAFT/miniemulsion styrene homopolymerizations were conducted using a tubular reactor. Controlled polymerizations were demonstrated through the use of a multi-tube reactor allowing five different residence times (conversions) to be studied at once. Molecular weights were shown to be close to theoretical predictions and the polydispersity of the polymer formed in the tube reactor was higher than that formed in batch. The ability of the dormant chains to continue adding monomer to form copolymer was demonstrated by a chain extension experiment with *n*-butyl acrylate. Modified tracer experiments using an oil-soluble dye

revealed that even though the Reynolds number was exceedingly low (~ 10-20) during the polymerizations, the reactor flow regime was not laminar. This was explained in terms of slippage at the wall of the tubing. The axial dispersion was quantified from the tracer data and found to be quite high, ~ 0.1. Ideal flow was achieved by metering nitrogen into the reactor to form small, isolated plugs of miniemulsion. This was termed "isolated plug flow" and the ideality was verified by tracer. Polymerizations conducted in this isolated plug flow regime were shown to produce polymer with similar polydispersities to polymer formed in batch, verifying the relationship between the residence time distribution in the reactor and the polydispersity of the polymer formed using CRP in miniemulsion. Kinetic discrepancies between batch and tube in isolated plug flow were explained in terms of loss of the water soluble initiator in small droplets left behind by the miniemulsion plugs as they traversed the reactor. Substantiating evidence for this phenomenon was presented in the form of both experimental data and modeling.

5.2 Suggestions for further inquiry

It is most often the case in the scientific endeavor that the answer to one question leaves other questions in its wake. This investigation proved no different, with many new questions emerging along the way. What follows is an outline of three that are considered to be the most significant in terms of contributing to the future success of employing RAFT chemistry.

5.2.1 Limiting conversion in VA/MESA polymerization

Because the fundamental appeal of CRPs lies in their ability to synthesize unique architectures, i.e. copolymers, it is vitally important that a given system retain as many dormant chains as possible while reaching high conversions. This allows for the facile extension of the chains with a second monomer and insures that the copolymer formed will contain a relatively small number of dead chains. While it was demonstrated in this work that in bulk, with two particular xanthates, full conversions could be achieved in this manner, problems remain when attempting to employ the same recipe in miniemulsion. This could be overcome in large part, but not completely, by the use of water soluble initiators or simply higher levels of the oil-soluble initiator. Further, attempts to extend the chains of the polymer formed in miniemulsion failed. The results shown in Figure 5.1 may hold a clue to the reason for this failure. It shows the SEC/UV chromatograms (proportional to the number of C=S bonds and therefore the number of dormant chains) for a typical miniemulsion using AIBN. They reveal a loss of the C=S functionality, and by extension dormant chains, of the incorporated RAFT agent as the reaction progresses. In principle, in a well-behaved RAFT reaction, the population of dormant chains would increase rapidly at the beginning of the reaction and remain relatively steady with little loss to irreversibly cross-termination of the intermediate RAFT radical (Scheme 1.1, **5**).[1-3]

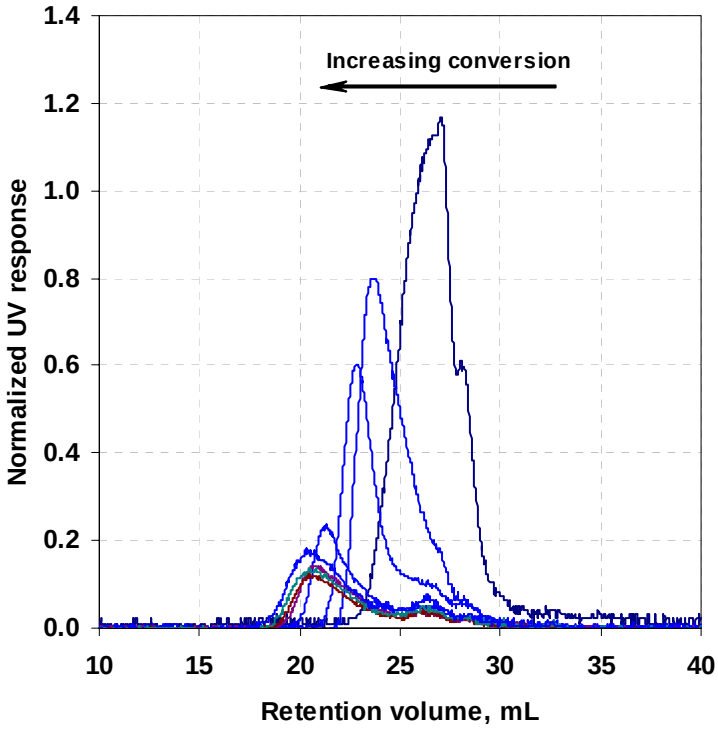


Figure 5.1 Normalized UV response of polymer samples from Exp 2. Shows loss of C=S bond of RAFT agent as reaction progresses, indicating termination and loss of dormant chains.

The questions to be addressed are as follows:

1. Does this evidence represent a significant enough loss of the C=S bond to account for the limiting conversion or should the experimental data be interpreted differently?

2. If not an artifact, then does the same phenomenon occur in bulk where the polymerizations achieve what is essentially complete conversion? If no, then what is the mechanism for the loss and what is it about conducting the polymerization in miniemulsion that either causes or exacerbates the loss? If yes, then it is not directly related to the limiting conversion and the source lies elsewhere. Also if yes, though not responsible for the limiting conversion, it may still be the fundamental reason for the inability to extend the chains.

3. Is it the direct cause of the limiting conversion or simply a symptom of some underlying event that is caused or aggravated by conducting in miniemulsion, e.g. intermediate radical termination?

The following recommendations are offered as a means to sort out the answers to these questions. As to the first question, there are two ways in which the experimental data might be interpreted differently. The first is that what is observed may be quite natural for the system employed. There is always some probability that the intermediate will terminate robbing the system of dormant chains. The extent to which that occurs is difficult to predict and depends on the relative rate constants involved. (It is still the subject of much debate within the scientific community.) This will be quite different for any given system and it may well be the case that what is observed is normal for this particular set of compounds. The other way in which the data might be interpreted follows from the manner in which the data were collected. As pointed out in Chapter 3, when employed to analyze single end groups, as is the case here, the distribution generated by the UV signal represents a cumulative number distribution, $N(M)$, not a mass distribution. In other words, the signal is proportional to the total number of end groups and by extension the total number of chains. The GPC distribution, $G(V_e)$, obtained via differential refractometry (RI) is a cumulative mass distribution and where the GPC calibration curve is linear the two are related by[4]:

$$N(M) = \mathrm{C} \times \frac{G(V_e)}{M^2}$$

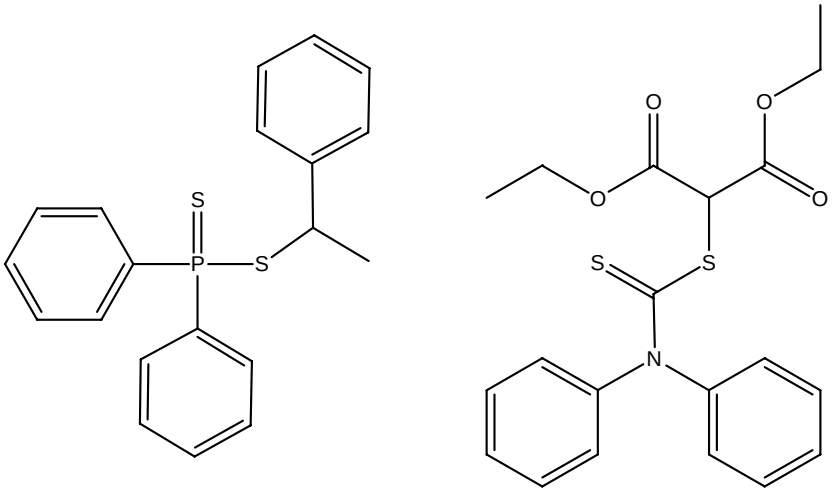
where M is the molecular weight of the polymer and C is an arbitrary constant which is a function of the calibration. As this relationship shows, the number distribution is an inverse function of the square of the molecular weight. This implies that as the molecular weight increases (as revealed in the GPC chromatogram), the concurrent magnitude of the number distribution (as revealed in the UV chromatogram) decreases by a factor of C/M^2. This assumes, of course, that the number distribution of total polymer chains is

perfectly represented by the UV response. The assumption is not exact when applied to the work presented here because the UV response represents the dormant chains only, not the entire chain population. However, since the number of dead chains in relatively low compared to the number of dormant chains as a percentage of the total number of chains, the analogy can be considered valid in a general conceptual sense. What all this implies in relation to the work presented here is that this innate relationship between the mass and number distribution may be responsible, either wholly or in part, for the observed diminution in the UV response of the C=S double bond and not an actual wholesale loss of functionality.

Given these two scenarios, the fundamental issue that must be sorted out is which one, if either, is correct. Comparing the UV chromatograms of bulk and miniemulsion experiments is important and easily undertaken first step. However, only if the two are significantly different can any firm conclusion be drawn. If the analysis suggests that the phenomenon is present in both, the question of which of the two scenarios that were presented holds cannot be answered without further investigation. In either event, the fate of C=S bonds and whether or not significant intermediate termination is occurring should be determined using alternate techniques. Two are suggested, both have been utilized with RAFT polymers and have proven useful. Since the carbon that forms the intermediate radical has its origin from carbon disulfide in the synthesis of MESA (as well as most RAFT dithioesters), $^{13}CS_2$ could easily be utilized in the synthesis, either exclusively or by simply mixing with an appropriate amount of CS_2. If termination of the intermediate radical occurs, the quaternary carbon formed can be easily tracked using ^{13}C NMR. Calitz et al.[5] employed this technique successfully to track the formation of

intermediate species in cumyl dithiobenzoate (CDB) mediated polymerizations of styrene. Another commonly employed technique to characterize the chain end groups in RAFT polymerizations is MALDI-TOF mass spectroscopy[6-9]. Taken together with the UV data, the results using these two techniques should establish the fate of the C=S bond and whether or not significant intermediate radical termination exists. If the termination is shown to be more prevalent in miniemulsion than in bulk, then a mechanism will have to be elucidated to account for the higher level.

If intermediate radical termination exists to no larger degree in miniemulsion than bulk then the source of the limiting conversion cannot be directly related to termination of the intermediate using MESA and therefore must be elsewhere. As such, these last three suggestions are offered only in this event and in the expectation that they would provide some clue as to the source and to point the research in that direction. In this case, solubility of vinyl acetate could be probed as a contributing factor. Vinyl hexanoate is structurally similar to vinyl acetate but several orders of magnitude less soluble in water[10,11]. While its polymerization rate is less than vinyl acetate, its transfer to monomer is similar[12-14]. There are other RAFT agents that have been shown to polymerize vinyl acetate in a controlled manner[6,15], two potential compounds are shown in Scheme 5.1. Both have been reported to offer a reasonable degree of control over VA bulk polymerizations but currently there are no reports in the open literature of their use in aqueous systems.



A **B**

Scheme 5.1 Proposed alternate RAFT agents for VA polymerization in miniemulsion: (**A**) diphenyl-phosphinodithioic acid 1-phenyl-ethyl ester, (**B**) 2-diphenyl-thiocarbamoylsulfanyl-malonic acid diethyl ester

5.2.2 Copolymers of VA using RAFT

As already mentioned, CRPs are currently most attractive as a means to create polymers of well-defined and unique architectures via radical chemistry. With this as the objective and as it relates to the work on vinyl acetate presented herein, it is crucial that it be demonstrated that this can be achieved using RAFT chemistry in an aqueous dispersed system. It follows that once the complications related to homopolymerizations of VA in miniemulsion are understood that the next logical step is to examine the synthesis of copolymers of VA using miniemulsions. Currently, progress in this arena may well be limited more by choice of a suitable RAFT agent than by any complications that might arise from employing miniemulsion as reaction medium. What makes a RAFT agent effective with VA, stabilization of the thiocarbonyl bond, tends to reduce its reactivity.

With a very reactive radical like VA this is an advantage and the main reason for the success of those RAFT agents that have been shown to control the polymerization of VA. This advantage is lost or even reversed with a monomer that forms a more stable radical, making the production of copolymers of VA difficult via RAFT techniques. That is not to say that it has not been achieved or that none of the currently available RAFT agents are capable of mediating copolymerizations of VA with another monomer. The field is just extremely limited at this point in time. The synthesis of a block copolymer of polystyrene and PVA has been reported by Gigmes et al.[15,16] using 1-phenylethyl-diphenylphosphinodithioate, an analog to **A** shown in Scheme 5.1. One report[17] claims to have produced poly(*n*-butyl acrylate)-*b*-poly(vinyl acetate) using 1-phenylethyl dithiobenzoate, a commonly used dithioester for styrene and acrylate controlled polymerizations. The carbamate **B** shown in Scheme 5.1 was demonstrated to control the polymerization of both VA and ethyl acrylate[6]. As such, basing the polymerization order on the most stable radical first, it may well be possible to form a copolymer of poly(ethyl acrylate)-*b*-poly(vinyl acetate). Coote et al.[18] recently reported detailed ab initio molecular orbital calculations that indicate that using fluorine as the Z-substituent (fluoro-dithioformates, S=C(F)SR, what they term F-RAFT agents) would provide improved control of very reactive monomers like VA without deactivating the C=S bond to the extent that more stable propagating radicals (e.g. styrene) will not add to the RAFT agent. No experimental evidence was presented to validate the assertion, however the authors mention that evaluation of F-RAFT agents was underway.

Star homopolymers of PVA have already been synthesized using RAFT and subsequently hydrolyzed to form star polymers of poly(vinyl alcohol)[19]. Copolymers of

PVA and a suitable comonomer open the door to hydrolyzing the PVA to PVOH to form a copolymer with hydrophilic/hydrophobic properties that could be exploited or tuned via architecture. A simple example would be block copolymer of PS and PVOH which might be employed as a surfactant.

5.2.3 CRP copolymerizations in continuous tube reactor

Finally, it remains to be demonstrated that continuous reactors, and in particular a continuous tubular reactor, can be successfully employed to make copolymers via CRP/miniemulsion while at the same time offering significant enough advantages over batch polymerization to be attractive. The obvious benefit of continuous systems is their ability to turn out large quantities of a very consistent product in an cost-effective fashion. This is largely a financial calculation and depends heavily upon the market economics of the particular polymer that is being manufactured. Currently, batch systems dominate world polymer production but competition is forcing consideration of continuous processes. But this benefit accrues to *any* reaction when conducted in a continuous system. What must be shown for CRP/miniemulsion systems is some benefit specific to the combination that cannot be achieved any other way. For single tubular reactors, this is problematic because in principle any polymer or copolymer that is synthesized in a tube reactor[§] could just as easily be synthesized in a batch process. However, continuous stirred reactors (CSTR) have the capacity to synthesize copolymers with a constant copolymer composition, without the natural composition drift that occurs in batch and tube reactors. By combining tube and stirred reactors, the potential to

[§] For the purposes of this argument, tube reactors in non-ideal flow, where there exists a significant residence time distribution, are not considered.

synthesize copolymers that cannot be made in batch systems could be realized. The question remains whether or not this would occur in a CRP/miniemulsion, where the polymerization locus is the particles which behave, in principle, as batch reactors. For example, were one to feed a miniemulsion comprised of one monomer into a stirred reactor which included a second, different monomer feed, a copolymer of constant composition would not be formed. The reason for this is that while the second monomer concentration in each particle would remain constant (because its overall concentration remains constant), the first would not (because its concentration would change within each particle as it polymerized). However, in some cases this limitation would not be an impediment. A tube reactor polymerizing one monomer, **A**, to complete conversion feeding a stirred reactor copolymerizing via two additional monomer feeds (could be **A** and **B** or **B** and **C**) is only one such example. Finally, the effects of monomer solubilities, diffusion rates, and particle morphologies will all need to be addressed in order to more fully understand the nature of employing CRP/miniemulsion combinations in continuous systems to form copolymers.

5.3 References

1. Barner-Kowollik, C.; Quinn, J. F.; Morsley, D. R.; Davis, T. P. *J. Polym. Sci. Polym. Chem.* **2001**, *39*, 1353.

2. Wang, A. R.; Zhu, S. P. *J. Polym. Sci. Polym. Chem.* **2003**, *41*, 1553.

3. Zhang, M.; Ray, W. H. *Ind. Eng. Chem. Res.* **2001**, *40*, 4336.

4. Gilbert, R. G., *Emulsion Polymerization : A Mechanistic Approach*. Academic Press: 1995.

5. Calitz, F. M.; McLeary, J. B.; McKenzie, J. M.; Tonge, M. P.; Klumperman, B.; Sanderson, R. D. *Macromolecules* **2003**, *36*, 9687.

6. Destarac, M.; Charmot, D.; Franck, X.; Zard, S. Z. *Macromol. Rapid Comm.* **2000**, *21*, 1035.

7. Ganachaud, F.; Monteiro, M. J.; Gilbert, R. G.; Dourges, M. A.; Thang, S. H.; Rizzardo, E. *Macromolecules* **2000**, *33*, 6738.

8. Mellon, W.; Rinaldi, D.; Bourgeat-Lami, E.; D'Agosto, F. *Macromolecules* **2005**, *38*, 1591.

9. Schilli, C.; Lanzendorfer, M. G.; Muller, A. H. E. *Macromolecules* **2002**, *35*, 6819.

10. El-Aasser, M. S.; Vanderhoff, J. W., *Emulsion Polymerization of Vinyl Acetate*. Applied Science Pub.: London, 1981.

11. Vanzo, E.; Marchessault, R. H.; Stannett, V. *J. Coll. Sci.* **1965**, *20*, 62.

12. Kurian, C. J.; Muthana, M. S. *Makromolekul. Chem.* **1959**, *29*, 1.

13. Ham, G. E., *Kinetics and Mechanisms of Polymerization*. M. Dekker: New York, NY, 1967; Vol. 1.

14. Kurian, C. J.; Muthana, M. S. *Makromolekul. Chem.* **1959**, *29*, 19.

15. Gigmes, D.; Bertin, D.; Marque, S.; Guerret, O.; Tordo, P. *Tetrahedron Lett.* **2003**, *44*, 1227.

16. Bertin, D.; Gigmes, D.; Guerret, O.; Tordo, P.; Vuillemin, B. WO 02/096955, 2002.

17. Zhuang, R. C.; Chen, H. H.; Lin, J.; Ye, J. L.; Zou, Y. S. *Acta Polym. Sin.* **2001**, 288.

18. Coote, M. L.; Henry, D. J. *Macromolecules* **2005**, *38*, 5774.

19. Stenzel, M. H.; Davis, T. P.; Barner-Kowollik, C. *Chem. Commun.* **2004**, 1546.

APPENDIX

FORTRAN CODE FOR COMPUTER SIMULATIONS

A.1 Laminar flow in tubular reactor

```
Program pdi_tube
implicit none

integer         i,j,n,interv,check
integer         nok,nbad


double precision          param(50)
double precision          t,t0,tf
double precision          yini(23),y(23),yscal(23),dydx(23)
double precision          M0,I0,tau,theta,f
double precision          func,errest
double precision          Xpraft,Xpout
double precision          PDraft,PDtube,PDout
double precision          momnt2,momnt1,momnt0
double precision          mmnt2out,mmnt1out,mmnt0out
double precision          kd,kp,ktc,ktd,kct,kt
double precision          ktrCTA,ktrm,ktrs,kraftf,kraftr
double precision          Cm,Car,Ccta,Cs,Cr,Crar,Cini
double precision          Temp, R
double precision          hdid,hnext,htry,hmin
double precision          eps
double precision          Xp,DPn,DPw,Zp

c     *** Common blocks

common /b1/eps,htry
common /b2/tau
common /b3/yini
common /b4/y
common /ratecons/kd,kp,ktc,ktd,kct,ktrCTA,ktrm,
ktrs,kraftf,kraftr,f
common /concentrations/Cm,Car,Ccta,Cs,Cr,Crar,Cini

c     *** Function declarations

external derivs,midinf,stifbs,jacobn
external PDtube,XPtube,PDraft
```

```
external momnt2tube,momnt1tube,momnt0tube

c    *** Defines constants.

R=0.008314d0             ! Universal gas constant, kJ/mol*K
Temp=60.0d0              ! Temperature in °C
Temp=Temp+273.15d0       ! Temperature in Kelvin

c    *** Set initial concentrations

Cm=8.685d0     ! Initial monomer concentration, mol/L
Cs=0.0d0       ! Initial solvent concentration, mol/L
Car=Cm/750.0d0 ! Initial RAFT concentration, mol/L
Cini=Car/1.0d0 ! Initial initiator concentration, mol/L
Cr=0.0d0       ! Initial primary radical concentration,
               ! mol/L
Ccta=0.0d0     ! Initial chain transfer agent
               ! concentration, mol/L
Crar=0.0d0     ! Initial primary intermediate
               ! concentration, mol/L
f=0.65d0       ! Initiator efficiency


c    *** Rate constants for styrene/PEPDTA system ***
c    *** Initiator (AIBN) decomposition constant, 1/s

kd=1.29d15*dexp(-127.6d0/(R*Temp))

c    *** Rate constant of propagation for styrene,
L/(mol*s)
c    *** (Macromol. Chem. Phys. 1995,196,3267)

     kp=10.0**7.63*dexp(-32.51d0/(R*Temp))

c    *** Forward RAFT reaction rate constant, L/(mol*s)
c    *** (See Macromolecules,2002,35,7582)

     kraftf=5.6d5

c    *** Reverse RAFT reaction rate constant, 1/s
c    *** (See Macromolecules,2002,35,7582)

     kraftr= 2.7d-1

c    *** Rate constant of termination by combination

     ktc=1.703d9*dexp(-2268.d0/(1.987*Temp))
```

```
c     *** Rate constant of termination by disproportionation

      ktd=0.0d0

c     *** Rate constant, transfer to monomer

      ktrm=6.0d-5

c     *** Rate constant, transfer to solvent

      ktrs=0.0d0

c     Rate constant, transfer to CTA, assumes all RAFT agent
      added to monomer (or Ct is very high)

      ktrCTA=0.0d0


c     *** Set initial conditions

      yini(1)=0.0d0
      yini(2)=0.0d0
      yini(3)=0.0d0
      yini(4)=0.0d0
      yini(5)=0.0d0
      yini(6)=0.0d0
      yini(7)=0.0d0
      yini(8)=0.0d0
      yini(9)=0.0d0
      yini(10)=0.0d0
      yini(12)=0.0d0
      yini(13)=0.0d0
      yini(14)=0.0d0
      yini(15)=0.0d0
      yini(16)=0.0d0
      yini(17)=Cr
      yini(18)=Car
      yini(19)=Crar
      yini(20)=Cini
      yini(21)=Cm
      yini(22)=Cs
      yini(23)=Ccta

c        *** Opens output files.

      open(unit=1,file='Int1.out',status='unknown')
      open(unit=2,file='Int2.out',status='unknown')
```

```
      open(unit=3,file='Int3.out',status='unknown')
      open(unit=4,file='Int4.out',status='unknown')
      open(unit=5,file='Derivs.out',status='unknown')

      n=23
      htry=1.0d-12
      eps=1.0d-6


      write(*,25)
25    format('Enter 1 for batch, 2 for tube: ',\)
      read*,check

      if (check.eq.1) then
      goto 100
      else
      goto 200
      endif

100   do i=1,5000
      y=yini
      t0=0.0d0
      tf=float(i)*100.0d0

      call odeint(y,n,t0,tf,eps,htry,hmin,nok,
      #nbad,derivs,stifbs)
      Xp=(y(2)+y(5)+y(8)+y(15)+y(11))/(y(2)+y(5)
      #+y(8)+y(15)+y(11)#+y(21))
      DPn=(y(2)+y(5)+y(8)+y(11)+y(15))/(y(1)+y(4)
      #+y(7)+0.5d0*y(10)+y(14))
      DPw=(y(3)+y(6)+y(9)+y(12)+0.5d0*y(13)+y(16))/
      #(y(2)+y(5)+y(8)+y(15)+y(11))
      Zp=DPw/DPn


      write(*,15)tf/3600.0,Xp,-log(1.d0-Xp),DPn,Dpw,Zp
      write(2,15)tf/3600.0,Xp,-log(1.d0-Xp),DPn,Dpw,Zp

      if (Xp.gt.0.99d0) goto 1
      enddo
      goto 1

c     *** This loop performs the integration of the laminar
      flow profile.

200   do i=1,100
```

```
c     *** Sets limits of integration

      tau=dfloat(i)*10000.0d0
      t0=tau/2.0d0
      tf=1.0d20

c     *** Function calls to calculate moments

      call qromo(Xptube,t0,tf,Xpout,midinf)
      call qromo(momnt2tube,t0,tf,mmnt2out,midinf)
      call qromo(momnt1tube,t0,tf,mmnt1out,midinf)
      call qromo(momnt0tube,t0,tf,mmnt0out,midinf)

      PDout=(mmnt2out*mmnt0out/mmnt1out**2)

      write(*,30)tau/3600.d0,Xpout,PDout
      write(1,30)tau/3600.d0,Xpout,PDout

      y=yini

      call odeint(y,n,0.0d0,tau,eps,htry,hmin,nok,nbad,
     #derivs,stifbs)
      Xp=(y(2)+y(5)+y(8)+y(15)+y(11))/(y(2)+y(5)
     #+y(8)+y(15)+y(11)+y(21))
      DPn=(y(2)+y(5)+y(8)+y(11)+y(15))/(y(1)+y(4)
     #+y(7)+0.5d0*y(10)+y(14))
      DPw=(y(3)+y(6)+y(9)+y(12)+0.5d0*y(13)+y(16))/
     #(y(2)+y(5)+y(8)+y(15)+y(11))
      Zp=DPw/DPn

      write(*,30)tau/3600.d0,Xp,Zp
      write(2,30)tau/3600.d0,Xp,Zp

      enddo
      goto 1

10    format(1x,f8.3,1x,f8.6,1x,f10.7,1x,f8.6,1x,f10.7)
15    format(1x,f8.3,1x,f10.6,1x,f10.6,1x,f12.6,1x,f12.6,
     #1x,f10.7)
20    format(1x,e16.5,1x,e16.5,1x,e16.5,1x,e16.5,1x,e16.5)
30    format(1x,f8.1,1x,f8.6,1x,f16.12,1x,f10.7)

1     pause 'Press any key to continue'
      end

c     **************Subroutines and functions.*********
c     ***********************************************
```

```
c          **************FUNCTIONS***************
c          ************************************

      double precision function momnt2(t)
      implicit none
      integer i,n,nok,nbad
      double precision         t,t0,tf,tau
      double precision         yt(23),yini(23)
      double precision         eps,htry,hmin
      double precision         DPn,DPw,Zp
      common /b1/eps,htry
      common /b2/tau
      common /b3/yini

      external derivs,stifbs
      n=23
      yt=yini
      hmin=0.0d0
      t0=0.0d0
      tf=t
      call
odeint(yt,n,t0,tf,eps,htry,hmin,nok,nbad,derivs,stifbs)

      momnt2=yt(3)+yt(6)+yt(9)+yt(12)+0.5d0*yt(13)+yt(16)

      return
      end

c     ******************

      double precision function momnt1(t)
      implicit none
      integer i,n,nok,nbad
      double precision         t,t0,tf,tau
      double precision         yt(23),yini(23)
      double precision         eps,htry,hmin
      double precision         DPn,DPw,Zp
      common /b1/eps,htry
      common /b2/tau
      common /b3/yini

      external derivs,stifbs
      n=23
      yt=yini
      hmin=0.0d0
      t0=0.0d0
```

```
      tf=t
      call
odeint(yt,n,t0,tf,eps,htry,hmin,nok,nbad,derivs,stifbs)

      momnt1=yt(2)+yt(5)+yt(8)+yt(11)+yt(15)

      return
      end

c     ******************

      double precision function momnt0(t)
      implicit none
      integer i,n,nok,nbad
      double precision         t,t0,tf,tau
      double precision         yt(23),yini(23)
      double precision         eps,htry,hmin
      double precision         DPn,DPw,Zp
      common /b1/eps,htry
      common /b2/tau
      common /b3/yini

      external derivs,stifbs
      n=23
      yt=yini
      hmin=0.0d0
      t0=0.0d0
      tf=t
      call
odeint(yt,n,t0,tf,eps,htry,hmin,nok,nbad,derivs,stifbs)

      momnt0=yt(1)+yt(4)+yt(7)+0.5d0*yt(10)+yt(14)

      return
      end

c     ******************

      double precision function momnt2tube(t)
      implicit none

      double precision         t,tau,momnt2
      common /b2/tau

      momnt2tube=momnt2(t)*tau**2/(2.0d0*t**3)

      return
```

```
      end

c     ********************

      double precision function momnt1tube(t)
      implicit none

      double precision         t,tau,momnt1
      common /b2/tau
      momnt1tube=momnt1(t)*tau**2/(2.0d0*t**3)

      return
      end

c     ********************

      double precision function momnt0tube(t)
      implicit none

      double precision         t,tau,momnt0
      common /b2/tau
      momnt0tube=momnt0(t)*tau**2/(2.0d0*t**3)

      return
      end

c     ********************

      double precision function PDtube(t)
      implicit none

      double precision         t,t0,tf,tau
      double precision         eps,htry,hmin
      double precision         DPn,DPw,Zp,PDraft
      common /b1/eps,htry
      common /b2/tau

      PDtube=PDraft(t)/t**3

      return
      end

c     ******************

      double precision function PDraft(t)
      implicit none
      integer i,n,nok,nbad
```

```
      double precision         t,t0,tf,tau
      double precision         yt(23),yini(23)
      double precision         eps,htry,hmin
      double precision         DPn,DPw,Zp
      common /b1/eps,htry
      common /b2/tau
      common /b3/yini

      external derivs,stifbs
      n=23
      yt=yini
      hmin=0.0d0
      t0=0.0d0
      tf=t
      call
odeint(yt,n,t0,tf,eps,htry,hmin,nok,nbad,derivs,stifbs)


      DPn=(yt(2)+yt(5)+yt(8)+yt(11)+yt(15))/(yt(1)+yt(4)+yt(
7)+0.5d0*
      #yt(10)+yt(14))
      DPw=(yt(3)+yt(6)+yt(9)+yt(12)+0.5d0*yt(13)+yt(16))/
      #(yt(2)+yt(5)+yt(8)+yt(15)+yt(11))
      Zp=DPw/DPn

      PDraft=Zp

      return
      end

c     ******************

      double precision  function Xpraft(t)
      implicit none

      integer n,nok,nbad
      double precision         t,t0,tf
      double precision         yt(23),yini(23)
      double precision         eps,htry,hmin
      double precision         Xp,DPn
      common /b1/eps,htry
      common /b3/yini

      external derivs,stifbs

      n=23
      yt=yini
```

```
      hmin=0.0d0
      t0=0.0d0
      tf=t

      call
odeint(yt,n,t0,tf,eps,htry,hmin,nok,nbad,derivs,stifbs)

      DPn=(yt(2)+yt(5)+yt(8)+yt(11)+yt(15))/(yt(1)+yt(4)+yt(
7)+
     #0.5*yt(10)+yt(14))
      Xp=(yt(2)+yt(5)+yt(8)+yt(15)+yt(11))/(yt(2)+yt(5)+yt(8
)+
     #yt(15)+yt(11)+yt(21))
      Xpraft=Xp

      return
      end

c     ******************

      double precision  function XPtube(t)
      implicit none
      double precision t,tau
      double precision Xpraft

      common /b2/tau

      Xptube=Xpraft(t)*tau**2/(2.0d0*t**3)

      return
      end


c          **************SUBROUTINES****************
c          ***************************************

      subroutine derivs(x,y,dydx)
      implicit none

c     *** Returns the derivatives defining the system

      double precision         x,y(*),dydx(*)
      double precision         kd,kp,ktc,ktd,kct,f
      double precision        ktrCTA,ktrm,ktrs,kraftf,kraftr


      common /ratecons/kd,kp,ktc,ktd,kct,ktrCTA,ktrm,ktrs,
```

```
     #kraftf,kraftr,f


     dydx(1)=kp*y(17)*y(21)-(ktc+ktd)*(y(17)+y(1))*y(1)-
ktrs*
     #y(1)*y(22)-ktrCTA*y(1)*y(23)-
kraftf*y(1)*(y(18)+y(7))+
     #kraftr*(y(4)+y(10))

     dydx(2)=kp*y(17)*y(21)+ktrm*y(1)*y(21)+kp*y(21)*y(1)-
     #(ktc+ktd)*(y(17)+y(1))*y(2)-ktrm*y(2)*y(21)-ktrs*
     #y(2)*y(22)-ktrCTA*y(2)*y(23)-kraftf*y(2)*(y(18)+
     #y(7))+kraftr*(y(5)+y(11))

     dydx(3)=kp*y(17)*y(21)+kp*y(21)*(y(1)+2.0d0*y(2))-
(ktc+ktd)*
     #(y(17)+y(1))*y(3)-ktrm*y(3)*y(21)-ktrs*y(3)*y(22)-
ktrCTA*y(3)*
     #y(23)-kraftf*y(3)*(y(18)+y(7))+kraftr*(y(6)+y(12))

     dydx(4)=kraftf*y(1)*y(18)+kraftf*y(17)*y(7)-
2.0d0*kraftr*y(4)
     dydx(5)=kraftf*y(2)*y(18)+kraftf*y(17)*y(8)-
2.0d0*kraftr*y(5)
     dydx(6)=kraftf*y(3)*y(18)+kraftf*y(17)*y(9)-
2.0d0*kraftr*y(6)

     dydx(7)=-kraftf*(y(17)+y(1))*y(7)+kraftr*(y(4)+y(10))
     dydx(8)=-kraftf*(y(17)+y(1))*y(8)+kraftr*(y(5)+y(11))
     dydx(9)=-kraftf*(y(17)+y(1))*y(9)+kraftr*(y(6)+y(12))

     dydx(10)=2.0d0*kraftf*y(1)*y(7)-2.0d0*kraftr*y(10)
     dydx(11)=kraftf*(y(2)*y(7)+y(1)*y(8))-
2.0d0*kraftr*y(11)
     dydx(12)=kraftf*(y(3)*y(7)+y(1)*y(9))-
2.0d0*kraftr*y(12)
     dydx(13)=2.0d0*kraftf*y(2)*y(8)-2.0d0*kraftr*y(13)

     dydx(14)=ktd*y(1)*(y(17)+y(1))+0.5d0*ktc*y(1)**2+ktc*y
(1)*y(17)
     #+ktrm*y(1)*y(21)+ktrs*y(1)*y(22)+ktrCTA*y(1)*y(23)

     dydx(15)=(ktc+ktd)*(y(17)+y(1))*y(2)+ktrm*y(2)*y(21)+k
trs*y(2)*
     #y(22)+ktrCTA*y(2)*y(23)
```

```
     dydx(16)=(ktc+ktd)*(y(17)+y(1))*y(3)+ktc*y(2)**2+ktrm*
y(3)*y(21)
     #+ktrs*y(3)*y(22)+ktrCTA*y(3)*y(23)

     dydx(17)=2.0d0*f*kd*y(20)-kp*y(17)*y(21)-
(ktc+ktd)*y(17)*(y(17)+
     #y(1))+ktrs*y(1)*y(22)+ktrCTA*y(1)*y(23)-
kraftf*y(17)*(y(18)+
     #y(7))+kraftr*(2.0d0*y(19)+y(4))

     dydx(18)=-
kraftf*y(18)*(y(17)+y(1))+kraftr*(2.0d0*y(19)+y(4))
     dydx(19)=kraftf*y(17)*y(18)-2.0d0*kraftr*y(19)
     dydx(20)=-kd*y(20)
     dydx(21)=-kp*y(21)*(y(17)+y(1))-ktrm*y(1)*y(21)
     dydx(22)=-ktrs*y(1)*y(22)
     dydx(23)=-ktrCTA*y(1)*y(23)

     return
     end

c    ******************

     subroutine jacobn(x,y,dfdx,dfdy,n,nmax)
     implicit none

c    Returns the Jacobian of the system of differential
equations.
c    Used by stifbs

     integer                        i,j,n,nmax
     double precision
     x,y(*),dfdx(*),dfdy(nmax,nmax)
     double precision         kd,kp,ktc,ktd,kct,f
     double precision        ktrCTA,ktrm,ktrs,kraftf,kraftr

     common /ratecons/kd,kp,ktc,ktd,kct,ktrCTA,ktrm,ktrs,
     #kraftf,kraftr,f

     do i=1,n
     dfdx(i)=0.0d0
     enddo

     dfdy(1,1) = -(ktc+ktd)*y(1)-(ktc+ktd)*(y(17)+y(1))-
ktrs*y(22)-
     #ktrCTA*y(23)-kraftf*(y(18)+y(7))
     dfdy(1,4) = kraftr
```

```
          dfdy(1,7) = -kraftf*y(1)
          dfdy(1,10) = kraftr
          dfdy(1,17) = kp*y(21)-(ktc+ktd)*y(1)
          dfdy(1,18) = -kraftf*y(1)
          dfdy(1,21) = kp*y(17)
          dfdy(1,22) = -ktrs*y(1)
          dfdy(1,23) = -ktrCTA*y(1)

          dfdy(2,1) = ktrm*y(21)+kp*y(21)-(ktc+ktd)*y(2)
          dfdy(2,2) = -(ktc+ktd)*(y(17)+y(1))-ktrm*y(21)-
ktrs*y(22)-ktrCTA*
          #y(23)-kraftf*(y(18)+y(7))
          dfdy(2,5) = kraftr
          dfdy(2,7) = -kraftf*y(2)
          dfdy(2,11) = kraftr
          dfdy(2,17) = kp*y(21)-(ktc+ktd)*y(2)
          dfdy(2,18) = -kraftf*y(2)
          dfdy(2,21) = kp*y(17)+ktrm*y(1)+kp*y(1)-ktrm*y(2)
          dfdy(2,22) = -ktrs*y(2)
          dfdy(2,23) = -ktrCTA*y(2)

          dfdy(3,1) = kp*y(21)-(ktc+ktd)*y(3)
          dfdy(3,2) = 2.0d0*kp*y(21)
          dfdy(3,3) = -(ktc+ktd)*(y(17)+y(1))-ktrm*y(21)-
ktrs*y(22)-ktrCTA*
          #y(23)-kraftf*(y(18)+y(7))
          dfdy(3,6) = kraftr
          dfdy(3,7) = -kraftf*y(3)
          dfdy(3,12) = kraftr
          dfdy(3,17) = kp*y(21)-(ktc+ktd)*y(3)
          dfdy(3,18) = -kraftf*y(3)
          dfdy(3,21) = kp*y(17)+kp*(y(1)+2.0d0*y(2))-
ktrm*y(3)
          dfdy(3,22) = -ktrs*y(3)
          dfdy(3,23) = -ktrCTA*y(3)

          dfdy(4,1) = kraftf*y(18)
          dfdy(4,4) = -2.0d0*kraftr
          dfdy(4,7) = kraftf*y(17)
          dfdy(4,17) = kraftf*y(7)
          dfdy(4,18) = kraftf*y(1)

          dfdy(5,2) = kraftf*y(18)
          dfdy(5,5) = -2.0d0*kraftr
          dfdy(5,8) = kraftf*y(17)
          dfdy(5,17) = kraftf*y(8)
          dfdy(5,18) = kraftf*y(2)
```

```
          dfdy(6,3) = kraftf*y(18)
      dfdy(6,6) = -2.0d0*kraftr
          dfdy(6,9) = kraftf*y(17)
          dfdy(6,17) = kraftf*y(9)
          dfdy(6,18) = kraftf*y(3)

          dfdy(7,1) = -kraftf*y(7)
          dfdy(7,4) = kraftr
          dfdy(7,7) = -kraftf*(y(17)+y(1))
          dfdy(7,10) = kraftr
          dfdy(7,17) = -kraftf*y(7)

          dfdy(8,1) = -kraftf*y(8)
          dfdy(8,5) = kraftr
          dfdy(8,8) = -kraftf*(y(17)+y(1))
          dfdy(8,11) = kraftr
          dfdy(8,17) = -kraftf*y(8)

          dfdy(9,1) = -kraftf*y(9)
          dfdy(9,6) = kraftr
          dfdy(9,9) = -kraftf*(y(17)+y(1))
          dfdy(9,12) = kraftr
          dfdy(9,17) = -kraftf*y(9)

          dfdy(10,1) = 2.0d0*kraftf*y(7)
          dfdy(10,7) = 2.0d0*kraftf*y(1)
          dfdy(10,10) = -2.0d0*kraftr

          dfdy(11,1) = kraftf*y(8)
          dfdy(11,2) = kraftf*y(7)
          dfdy(11,7) = kraftf*y(2)
          dfdy(11,8) = kraftf*y(1)
          dfdy(11,11) = -2.0d0*kraftr

          dfdy(12,1) = kraftf*y(9)
          dfdy(12,3) = kraftf*y(7)
          dfdy(12,7) = kraftf*y(3)
          dfdy(12,9) = kraftf*y(1)
          dfdy(12,12) = -2.0d0*kraftr

          dfdy(13,2) = 2.0d0*kraftf*y(8)
          dfdy(13,8) = 2.0d0*kraftf*y(2)
          dfdy(13,13) = -2.0d0*kraftr

          dfdy(14,1) =
ktd*(y(17)+y(1))+ktd*y(1)+ktc*y(1)+ktc*y(17)+
          #ktrm*y(21)+ktrs*y(22)+ktrCTA*y(23)
```

```
          dfdy(14,17) = ktd*y(1)+ktc*y(1)
          dfdy(14,21) = ktrm*y(1)
          dfdy(14,22) = ktrs*y(1)
          dfdy(14,23) = ktrCTA*y(1)

          dfdy(15,1) = (ktc+ktd)*y(2)
          dfdy(15,2) =
(ktc+ktd)*(y(17)+y(1))+ktrm*y(21)+ktrs*y(22)+ktrCTA*
          #y(23)
          dfdy(15,17) = (ktc+ktd)*y(2)
          dfdy(15,21) = ktrm*y(2)
          dfdy(15,22) = ktrs*y(2)
          dfdy(15,23) = ktrCTA*y(2)

          dfdy(16,1) = (ktc+ktd)*y(3)
          dfdy(16,2) = 2.0d0*ktc*y(2)
          dfdy(16,3) =
(ktc+ktd)*(y(17)+y(1))+ktrm*y(21)+ktrs*y(22)+ktrCTA*
          #y(23)
          dfdy(16,17) = (ktc+ktd)*y(3)
          dfdy(16,21) = ktrm*y(3)
          dfdy(16,22) = ktrs*y(3)
          dfdy(16,23) = ktrCTA*y(3)

          dfdy(17,1) = -
(ktc+ktd)*y(17)+ktrs*y(22)+ktrCTA*y(23)
          dfdy(17,4) = kraftr
          dfdy(17,7) = -kraftf*y(17)
          dfdy(17,17) = -kp*y(21)-(ktc+ktd)*(y(17)+y(1))-
(ktc+ktd)*y(17)-
          #kraftf*(y(18)+y(7))
          dfdy(17,18) = -kraftf*y(17)
          dfdy(17,19) = 2.0d0*kraftr
          dfdy(17,20) = 2.0d0*f*kd
          dfdy(17,21) = -kp*y(17)
          dfdy(17,22) = ktrs*y(1)
          dfdy(17,23) = ktrCTA*y(1)

          dfdy(18,1) = -kraftf*y(18)
          dfdy(18,4) = kraftr
          dfdy(18,17) = -kraftf*y(18)
          dfdy(18,18) = -kraftf*(y(17)+y(1))
          dfdy(18,19) = 2.0d0*kraftr

          dfdy(19,17) = kraftf*y(18)
          dfdy(19,18) = kraftf*y(17)
          dfdy(19,19) = -2.0d0*kraftr
```

```
          dfdy(20,20) = -kd

          dfdy(21,1) = -kp*y(21)-ktrm*y(21)
          dfdy(21,17) = -kp*y(21)
          dfdy(21,21) = -kp*(y(17)+y(1))-ktrm*y(1)

          dfdy(22,1) = -ktrs*y(22)
          dfdy(22,22) = -ktrs*y(1)

          dfdy(23,1) = -ktrCTA*y(23)
          dfdy(23,23) = -ktrCTA*y(1)

      return
      end

c     The following subroutines are modified from code
      available from NUMBERICAL RECIPES IN FORTRAN 77: THE
      ART OF SCIENTIFIC COMPUTING, 2nd ed., Vol. 1

      subroutine
      odeint(ystart,nvar,x1,x2,eps,h1,hmin,nok,nbad,derivs,
           #rkqs)
      integer nbad,nok,nvar,kmaxx,maxstp,nmax
      double precision eps,h1,hmin,x1,x2,ystart(nvar),tiny
      external derivs,rkqs
      parameter (maxstp=100000,nmax=50,kmaxx=200,tiny=1.d-
#30)

c     RUNGE-KUTTA DRIVER WITH ADAPTIVE STEPSIZE CONTROL

      integer i,kmax,kount,nstp
      double precision dxsav,h,hdid,hnext
      double precision x,xsav,dydx(nmax),xp(kmaxx),
           #y(nmax),yp(nmax,kmaxx),yscal(nmax)
      common /path/ kmax,kount,dxsav,xp,yp
c     user storage for intermediate results. preset dxsav
      and kmax.
      x=x1
      h=sign(h1,x2-x1)
      nok=0
      nbad=0
      kount=0
      do  i=1,nvar
      y(i)=ystart(i)
      enddo
      if (kmax.gt.0) xsav=x-2.*dxsav
```

```
      do  nstp=1,maxstp
      call derivs(x,y,dydx)
      do  i=1,nvar
c     scaling used to monitor accuracy.
      yscal(i)=abs(y(i))+abs(h*dydx(i))+tiny
      enddo
      if(kmax.gt.0)then
      if(abs(x-xsav).gt.abs(dxsav)) then
      if(kount.lt.kmax-1)then
      kount=kount+1
      xp(kount)=x
      do  i=1,nvar
      yp(i,kount)=y(i)
      enddo
      xsav=x
      endif
      endif
      endif
      if((x+h-x2)*(x+h-x1).gt.0.) h=x2-x
      call rkqs(y,dydx,nvar,x,h,eps,yscal,hdid,hnext,derivs)
      if(hdid.eq.h)then
      nok=nok+1
      else
      nbad=nbad+1
      endif
      if((x-x2)*(x2-x1).ge.0.)then
      do  i=1,nvar
      ystart(i)=y(i)
      enddo
      if(kmax.ne.0)then
      kount=kount+1
      xp(kount)=x
      do  i=1,nvar
      yp(i,kount)=y(i)
      enddo
      endif
      return
      endif
      if(abs(hnext).lt.hmin) pause
      #'stepsize smaller than minimum in odeint'
      h=hnext
      enddo
      pause 'too many steps in odeint'
      return
      end

c     ******************
```

```
      subroutine
      stifbs(y,dydx,nv,x,htry,eps,yscal,hdid,hnext,derivs)
      integer nv,nmax,kmaxx,imax
      double precision eps,hdid,hnext,htry
      double precision x,dydx(nv),y(nv),yscal(nv),
           #safe1,safe2,redmax,redmin,tiny,scalmx
      external derivs
      parameter (nmax=50,kmaxx=7,imax=kmaxx+1,
     #    safe1=.25,safe2=.7, redmax=1.d-5,redmin=.7,tiny=1.d-
     #    30,scalmx=.1)

c     USES DERIVS,JACOBN,SIMPR,PZEXTR
c     SEMI-IMPLICIT EXTRAPOLATION STEP FOR INTEGRATING STIFF
      O.D.E.'S, WITH MONITORING OF LOCAL TRUNCATION ERROR TO
      ADJUST STEPSIZE.

      integer i,iq,k,kk,km,kmax,kopt,nvold,nseq(imax)
      double precision eps1,epsold,errmax,fact,h
      double precision red,scale,work,wrkmin,
      #xest,xnew,a(imax),alf(kmaxx,kmaxx),
      #dfdx(nmax),dfdy(nmax,nmax),
      #err(kmaxx),yerr(nmax),ysav(nmax),yseq(nmax)
      logical first,reduct
      save a,alf,epsold,first,kmax,kopt,nseq,nvold,xnew
      data first/.true./,epsold/-1./,nvold/-1/
      data nseq /2,6,10,14,22,34,50,70/
      if(eps.ne.epsold.or.nv.ne.nvold)then
      hnext=-1.e29
      xnew=-1.e29
      eps1=safe1*eps
      a(1)=nseq(1)+1
      do  k=1,kmaxx
      a(k+1)=a(k)+nseq(k+1)
      enddo
      do  iq=2,kmaxx
      do  k=1,iq-1
      alf(k,iq)=eps1**((a(k+1)-a(iq+1))/
           #((a(iq+1)-a(1)+1.)*(2*k+1)))
      enddo
      enddo
      epsold=eps
      nvold=nv
      a(1)=nv+a(1)
      do  k=1,kmaxx
      a(k+1)=a(k)+nseq(k+1)
      enddo
```

```
      do  kopt=2,kmaxx-1
      if(a(kopt+1).gt.a(kopt)*alf(kopt-1,kopt))goto 1
      enddo
1     kmax=kopt
      endif
      h=htry
      do i=1,nv
      ysav(i)=y(i)
      enddo
      call jacobn(x,y,dfdx,dfdy,nv,nmax)
      if(h.ne.hnext.or.x.ne.xnew)then
      first=.true.
      kopt=kmax
      endif
      reduct=.false.
2     do k=1,kmax
      xnew=x+h
      if(xnew.eq.x)pause 'stepsize underflow in stifbs'
      call simpr(ysav,dydx,dfdx,dfdy,nmax,nv,
#     x,h,nseq(k),yseq,derivs)
      xest=(h/nseq(k))**2
      call pzextr(k,xest,yseq,y,yerr,nv)
      if(k.ne.1)then
      errmax=tiny
      do  i=1,nv
      errmax=max(errmax,abs(yerr(i)/yscal(i)))
      enddo
      errmax=errmax/eps
      km=k-1
      err(km)=(errmax/safe1)**(1./(2*km+1))
      endif
      if(k.ne.1.and.(k.ge.kopt-1.or.first))then
      if(errmax.lt.1.)goto 4
      if(k.eq.kmax.or.k.eq.kopt+1)then
      red=safe2/err(km)
      goto 3
      else if(k.eq.kopt)then
      if(alf(kopt-1,kopt).lt.err(km))then
      red=1./err(km)
      goto 3
      endif
      else if(kopt.eq.kmax)then
      if(alf(km,kmax-1).lt.err(km))then
      red=alf(km,kmax-1)*
           #safe2/err(km)
      goto 3
      endif
```

```
      else if(alf(km,kopt).lt.err(km))then
      red=alf(km,kopt-1)/err(km)
      goto 3
      endif
      endif
      enddo
3     red=min(red,redmin)
      red=max(red,redmax)
      h=h*red
      reduct=.true.
      goto 2
4     x=xnew
      hdid=h
      first=.false.
      wrkmin=1.d35
      do  kk=1,km
      fact=max(err(kk),scalmx)
      work=fact*a(kk+1)
      if(work.lt.wrkmin)then
      scale=fact
      wrkmin=work
      kopt=kk+1
      endif
      enddo
      hnext=h/scale
      if(kopt.ge.k.and.kopt.ne.kmax.and..not.reduct)then
      fact=max(scale/alf(kopt-1,kopt),scalmx)
      if(a(kopt+1)*fact.le.wrkmin)then
      hnext=h/fact
      kopt=kopt+1
      endif
      endif
      return
      end

c     ******************

      subroutine
      simpr(y,dydx,dfdx,dfdy,nmax,n,xs,htot,nstep,yout,
      #derivs)
      integer n,nmax,nstep,nmaxx
      double precision htot,xs,dfdx(n)
      double precision dfdy(nmax,nmax),dydx(n),y(n),
      #yout(n)
      external derivs
      parameter (nmaxx=50)
```

```
c     USES DERIVS,LUBKSB,LUDCMP
c     PERFORMS ONE STEP OF SEMI-IMPLICIT MIDPOINT RULE.

      integer i,j,nn,indx(nmaxx)
      double precision d,h,x,a(nmaxx,nmaxx),del(nmaxx)
      double precision ytemp(nmaxx)
      h=htot/nstep
      do i=1,n
      do j=1,n
      a(i,j)=-h*dfdy(i,j)
      enddo
      a(i,i)=a(i,i)+1.
      enddo
      call ludcmp(a,n,nmaxx,indx,d)
      do  i=1,n
      yout(i)=h*(dydx(i)+h*dfdx(i))
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
      do  i=1,n
      del(i)=yout(i)
      ytemp(i)=y(i)+del(i)
      enddo
      x=xs+h
      call derivs(x,ytemp,yout)
      do  nn=2,nstep
      do  i=1,n
      yout(i)=h*yout(i)-del(i)
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
      do  i=1,n
      del(i)=del(i)+2.*yout(i)
      ytemp(i)=ytemp(i)+del(i)
      enddo
      x=x+h
      call derivs(x,ytemp,yout)
      enddo
      do  i=1,n
      yout(i)=h*yout(i)-del(i)
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
      do  i=1,n
      yout(i)=ytemp(i)+yout(i)
      enddo
      return
      end

c     ******************
```

```
      subroutine ludcmp(a,n,np,indx,d)
      integer n,np,indx(n),nmax
      double precision d,a(np,np),tiny
      parameter (nmax=500,tiny=1.0d-20)

c     Given a matrix a(1:n,1:n), with physical dimension np
      by   np,   this   routine   replaces   it   by   the   LU
      decomposition of a rowwise permutation of itself.

      integer i,imax,j,k
      double precision aamax,dum,sum,vv(nmax)

      d=1.
      do  i=1,n
      aamax=0.0d0
      do  j=1,n
      if (abs(a(i,j)).gt.aamax) aamax=abs(a(i,j))
      enddo
      if (aamax.eq.0.) pause 'singular matrix in ludcmp'
      vv(i)=1./aamax
      enddo
      do  j=1,n
      do  i=1,j-1
      sum=a(i,j)
      do  k=1,i-1
      sum=sum-a(i,k)*a(k,j)
      enddo
      a(i,j)=sum
      enddo
      aamax=0.0
      do  i=j,n
      sum=a(i,j)
      do  k=1,j-1
      sum=sum-a(i,k)*a(k,j)
      enddo
      a(i,j)=sum
      dum=vv(i)*abs(sum)
      if (dum.ge.aamax) then
      imax=i
      aamax=dum
      endif
      enddo
      if (j.ne.imax)then
      do  k=1,n
      dum=a(imax,k)
      a(imax,k)=a(j,k)
```

```
      a(j,k)=dum
      enddo
      d=-d
      vv(imax)=vv(j)
      endif
      indx(j)=imax
      if(a(j,j).eq.0.0d0)a(j,j)=tiny
      if(j.ne.n)then
      dum=1./a(j,j)
      do  i=j+1,n
      a(i,j)=a(i,j)*dum
      enddo
      endif
      enddo
      return
      end

c     ******************

      subroutine lubksb(a,n,np,indx,b)
      integer n,np,indx(n)
      double precision a(np,np),b(n)

c     SOLVES THE SET OF N LINEAR EQUATIONS A · X = B.

      integer i,ii,j,ll
      double precision sum
      ii=0
      do  i=1,n
      ll=indx(i)
      sum=b(ll)
      b(ll)=b(i)
      if (ii.ne.0)then
      do  j=ii,i-1
      sum=sum-a(i,j)*b(j)
      enddo
      else if (sum.ne.0.) then
      ii=i
      endif
      b(i)=sum
      enddo
      do  i=n,1,-1
      sum=b(i)
      do  j=i+1,n
      sum=sum-a(i,j)*b(j)
      enddo
      b(i)=sum/a(i,i)
```

```
      enddo
      return
      end

c     ******************

      subroutine pzextr(iest,xest,yest,yz,dy,nv)
      integer iest,nv,imax,nmax
      double precision xest,dy(nv),yest(nv),yz(nv)
      parameter (imax=13,nmax=50)

c     Use polynomial extrapolation to evaluate nv functions
      at x = 0 by fitting a polynomial to a sequence of
      estimates with progressively smaller values x = xest,
      and corresponding function vectors yest(1:nv). This
      call is number iest in the sequence of calls.
      Extrapolated function values are output as yz(1:nv),
      and their estimated error is output as dy(1:nv).
      Parameters: maximum expected value of iest is imax; of
      nv is nmax.

      integer j,k1
      double precision delta,f1,f2,q,d(nmax)
      double precision qcol(nmax,imax),x(imax)
      save qcol,x
      x(iest)=xest
      do  j=1,nv
      dy(j)=yest(j)
      yz(j)=yest(j)
      enddo
      if(iest.eq.1) then
      do  j=1,nv
      qcol(j,1)=yest(j)
      enddo
      else
      do  j=1,nv
      d(j)=yest(j)
      enddo
      do  k1=1,iest-1
      delta=1.0/(x(iest-k1)-xest)
      f1=xest*delta
      f2=x(iest-k1)*delta
      do  j=1,nv
      q=qcol(j,k1)
      qcol(j,k1)=dy(j)
      delta=d(j)-q
      dy(j)=f1*delta
```

```
      d(j)=f2*delta
      yz(j)=yz(j)+dy(j)
      enddo
      enddo
      do  j=1,nv
      qcol(j,iest)=dy(j)
      enddo
      endif
      return
      end

c     ******************

      subroutine polint(xa,ya,n,x,y,dy)
      integer n,nmax
      double precision dy,x,y,xa(n),ya(n)
      parameter (nmax=10) !largest anticipated value of n.

C     given arrays xa and ya, each of length n, and given a
      value x, this routine returns a value y, and an error
      estimate dy. if p(x) is the polynomial of degree n - 1
      such that p(xai) = yai, i = 1, . . . , n, then the
      returned value y = p(x).

      integer i,m,ns
      double precision den,dif,dift,ho,hp,w,c(nmax),d(nmax)
      ns=1
      dif=abs(x-xa(1))
      do i=1,n
      dift=abs(x-xa(i))
      if (dift.lt.dif) then
      ns=i
      dif=dift
      endif
      c(i)=ya(i)
      d(i)=ya(i)
      enddo
      y=ya(ns)
      ns=ns-1
      do  m=1,n-1
      do  i=1,n-m
      ho=xa(i)-x
      hp=xa(i+m)-x
      w=c(i+1)-d(i)
      den=ho-hp
      if(den.eq.0.)pause 'failure in polint'
      den=w/den
```

```
      d(i)=hp*den
      c(i)=ho*den
      enddo
      if (2*ns.lt.n-m)then
      dy=c(ns+1)
      else
      dy=d(ns)
      ns=ns-1
      endif
      y=y+dy
      enddo
      return
      end

c     ******************

      subroutine qromo(func,a,b,ss,choose)
      integer jmax,jmaxp,k,km
      double precision a,b,func,ss,eps
      external func,choose
      parameter (eps=1.d-4, jmax=14, jmaxp=jmax+1, k=5,
#     km=k-1)

c     uses polint
c     Romberg integration on an open interval. Returns as ss
      the integral of the function func from a to b, using
      any specified integrating subroutine choose and
      Romberg's method. Normally choose will be an open
      formula, not evaluating the function at the endpoints.
      it is assumed that choose triples the number of steps
      on each call, and that its error series contains only
      even powers of the number of steps. The routines
      midpnt, midinf, midsql, midsqu, are possible choices
      for choose. the parameters have the same meaning as in
      qromb.

      integer j
      double precision dss,h(jmaxp),s(jmaxp)
      h(1)=1.
      do j=1,jmax
      call choose(func,a,b,s(j),j)
      if (j.ge.k) then
      call polint(h(j-km),s(j-km),k,0.d0,ss,dss)
      if (abs(dss).le.eps*abs(ss)) return
      endif
      s(j+1)=s(j)
      h(j+1)=h(j)/9.d0
```

```
      enddo
      pause 'too many steps in qromo'
      end

c     ******************

      subroutine midinf(funk,aa,bb,s,n)
      integer n
      double precision aa,bb,s,funk
      external funk
      integer it,j
      double precision a,b,ddel,del,sum,tnm,func,x
      func(x)=funk(1.0d0/x)/x**2
      b=1.d0/aa
      a=1.d0/bb
      if (n.eq.1) then
      s=(b-a)*func(0.5d0*(a+b))
      else
      it=3**(n-2)
      tnm=it
      del=(b-a)/(3.d0*tnm)
      ddel=del+del
      x=a+0.5d0*del
      sum=0.0d0
      do j=1,it
      sum=sum+func(x)
      x=x+ddel
      sum=sum+func(x)
      x=x+del
      enddo
      s=(s+(b-a)*sum/tnm)/3.d0
      endif
      return
      end
```

A.2 Droplet model in isolated plug flow

```
      program N2_plugs
      implicit none

      integer
      i,j,nvar,np,nd,nn,n,sum
      integer                              nok,nbad,func
      integer                              prnt
      double precision                     lr,lp,ld,lg,vel
      double precision                     kd,kt,kp,vp,vd,qd,f
      double precision
      i0,ip,ig,iini,vpi,ii
      double precision                     x0,xp1,xg,xini,xx
      double precision                     dydx(2), yout(2),
y(2)
      double precision,allocatable::
      didt(:),i1(:),iout(:)
      double precision,allocatable::       iplug(:),idrops(:)
      double precision,allocatable::       ipdum(:),iddum(:)
      double precision,allocatable::
      dxdt(:),x1(:),xout(:)
      double precision,allocatable::       xplug(:),xdrops(:)
      double precision,allocatable::       xpdum(:),xddum(:)
      double precision                     tau,ti,tf,tp,td,tg,
ttotal
      double precision
      pi,area,dia,root,vfactor

c     common blocks

      common /b1/qd,vp,kd
      common /b2/iini,xini
      common /b3/kt,kp,f
      common /b4/vpi
      common /b5/i0,x0
      common /b6/func

c     function declarations

      external p1der, pder,rk4,xg,stifbs

c     defines constants.
      prnt=1
      nn=25
```

```
      vfactor=1.0d0
      lr=6000.0d0
      lp=1.0d0
      lg=2.0d0
      vel=0.4d0
      kd=2.20d-5
      kp=250.0d0
      kt=6.0d7
      pi=3.14159265359d0
      dia=0.1651d0
      nvar=2
      ti=0.0d0
      f=0.65d0

c     calculates the number of plugs and droplets

      np=lr/(lp+lg)
      nd=np/nn

c     allocates array sizes

      allocate(didt(np+1),i1(np+1),iout(np+1))
      allocate(iplug(np+1),idrops(nd))
      allocate(ipdum(np+1),iddum(nd))
      allocate(dxdt(np+1),x1(np+1),xout(np+1))
      allocate(xplug(np+1),xdrops(nd))
      allocate(xpdum(np+1),xddum(nd))

c     calculates various parameters


      area=pi*(dia/2.0d0)**2
      vp=lp*area
      vd=(pi*(0.1651d0/2.0d0)**2)/100.0d0*vfactor
      ld=2.0d0*vd
      ld=ld/(4.0d0/3.0d0*pi)
      root=1.0d0/3.0d0
      ld=ld**(root)
      ld=2.0d0*ld
      tau=lr/vel
      tp=lp/vel
      td=ld/vel
      tg=lg/vel
      qd=vd/td


c     sets initial concentrations
```

```
      x0=0.0d0
      i0=1.0d0

      ii=i0*dexp(-kd*tau)
      xx=xg(i0,x0,tau)



      i1=0.0d0
      i1(1)=i0
      x1=0.0d0
      idrops=0.0d0
      iddum=idrops
      iplug=0.0d0
      ipdum=iplug
      iplug(1)=i0
      ipdum(1)=i0
      iout=0.0d0
      xplug=0.0d0
      xdrops=0.0d0
      xddum=xdrops
      xpdum=xplug

c     opens output file.

      open(unit=1,file='n2plugs.out',status='unknown')
      open(unit=2,file='check.out',status='unknown')

c     calculates the concentration of the first plug through
c     the reactor. also establishes location and
concentration
c     of droplets.

      plug=1.0
      if(nd.ne.0)then
           sum=0.0d0
           vpi=vp
           do i=1,np
                     if(mod(i,nn).eq.0) then

                          sum=sum+1

                          i1(i+1)=ig(i1(i),tg+tp-td)
                          x1(i+1)=xg(i1(i),x1(i),tg+tp-td)
                          y(1)=i1(i+1)
                          y(2)=x1(i+1)
```

```
                              func=2
                              call odeint(y,nvar,ti,td,1.0d-
8,1.0d-8,0.0d0,
     #                        nok,nbad,p1der,stifbs)
                              x1(i+1)=y(2)
                              i1(i+1)=y(1)

                              idrops(sum)=i1(i+1)

                              if(vpi.gt.vp-(vp/vd-1)*vd)then
                              vpi=vp-vd*sum
                              endif

                         else

                              i1(i+1)=ig(i1(i),tg+tp)
                              x1(i+1)=xg(i1(i),x1(i),tg+tp)

                         endif


               enddo
          endif


          iplug=i1
          xplug=x1
          ipdum=iplug
          xpdum=xplug
          iddum=idrops
          xddum=xdrops
          vpi=vp

          write(1,*)ii,xx
          write(1,*)1.0d0,i1(np+1),x1(np+1)

c         determines the concentration and conversion for each
of
c         the plugs in the reactor, pi, i > 1.


          do i=1,np/10

               sum=0
               if(nd.eq.0)then
                    do j=1,np
```

```
                    ipdum(j+1)=ig(ipdum(j),tg+tp)
                    write(1,*)ipdum(j+1)
               enddo
          else
               do j=1,np
                    vp=vpi
c                   if(j.le.nd)then
                    if(mod(j,nn).eq.0) then
c                   if(i.gt.np-nd)then
                         sum=sum+1
                         y(1)=ig(ipdum(j),tg-td)
                         y(2)=xg(ipdum(j),xpdum(j),tg-td)
                         iddum(sum)=ig(idrops(sum),tg-td)
                         iini=idrops(sum)

                         call odeint(y,nvar,ti,tp+td,1.0d-
8,1.0d-8,0.0d0,
     #                   nok,nbad,pder,stifbs)
                         ipdum(j+1)=y(1)
                         xpdum(j+1)=y(2)
                         iddum(sum)=ipdum(j+1)

                    else

                         ipdum(j+1)=ig(ipdum(j),tg+tp)

     xpdum(j+1)=xg(ipdum(j),xpdum(j),tg+tp)

                    endif


               enddo
          endif
          iplug=ipdum
          idrops=iddum
          xplug=xpdum
          xdrops=xddum
          if(mod(i,prnt).eq.0) then
c
     write(*,*)((tp+tg)*i+tau)/tau,iplug(np+1),xplug(np+1)

     write(1,*)((tp+tg)*i+tau)/tau,iplug(np+1),xplug(np+1)
          endif
     enddo

     end
```

```
c    ******************subroutines and
functions.************************


      double precision function ig(i0,tg)
      implicit none

      double precision i0,tg
      double precision qd,vp,kd

      common /b1/qd,vp,kd

      ig=i0*dexp(-kd*tg)

      return
      end

c     ********
      double precision function ip(i0,vpi,t)
      implicit none

      double precision i0,vpi,t
      double precision qd,vp,kd

      common /b1/qd,vp,kd

      ip=i0*dexp(-kd*t)*(vpi-qd*t)

      return
      end

c     ********
      double precision function xg(i0,x0,t)
      implicit none

      double precision t,func
      double precision i0,x0,xout
      double precision qd,vp,kd
      double precision iini,xini
      double precision kt,kp,f
      external func

      common /b1/qd,vp,kd
      common /b2/iini,xini
      common /b3/kt,kp,f

      iini=i0
```

```
      xini=x0

      call qsimp(func,0.0d0,t,xout)
      xg=1.0d0-(1.0d0-xini)*dexp(-xout)


      return
      end

c     **********

      subroutine pder(t,y,dydx)
      implicit none

      double precision    t,iini,xini
      double precision    dydx(2),y(2)
      double precision    qd,vp,kd
      double precision    kt,kp,f

      common /b1/qd,vp,kd
      common /b2/iini,xini
      common /b3/kt,kp,f

      dydx(1)=-(qd/vp+kd)*y(1)+qd/vp*iini
      dydx(2)=kp*(1.0d0-y(2))*((2.0d0*f*kd*y(1))/kt)**0.5d0



      return
      end

c     *********

      subroutine p1der(t,y,dydx)
      implicit none

      double precision    t
      double precision    dydx(2),y(2)
      double precision    qd,vp,kd
      double precision    kt,kp,f
      double precision    vpi

      common /b1/qd,vp,kd
      common /b3/kt,kp,f
      common /b4/vpi

      dydx(1)=-(qd/vpi+kd)*y(1)
```

```
      dydx(2)=kp*(1.0d0-y(2))*((2.0d0*f*kd*y(1))/kt)**0.5d0

      return
      end

c     ******************

      subroutine jacobn(x,y,dfdx,dfdy,n,nmax)
      implicit none

c     returns the jacobian of the system of differential
equations.
c     used by stifbs

      integer                        i,j,n,nmax,func
      double precision
      x,y(n),dfdx(n),dfdy(nmax,nmax)
      double precision          qd,vp,kd,vpi
      double precision          kt,kp,f


      common /b1/qd,vp,kd
      common /b3/kt,kp,f
      common /b4/vpi
      common /b6/func

      if(func.eq.1)then !jacobian for the gap
           dfdy(1,1) = -kd
           dfdy(1,2) = 0.0d0
           dfdy(2,1) = dsqrt(2.0d0)/2.0d0*kp*(1-y(2))
     #     /(f*kd*y(1)/kt)**0.5d0*f*kd/kt
           dfdy(2,2) = -
dsqrt(2.0d0)*kp*(f*kd*y(1)/kt)**0.5d0
      else if(func.eq.2)then !jacobian for droplet exit
           dfdy(1,1) = -kd-qd/vpi
           dfdy(1,2) = 0.0d0
           dfdy(2,1) = dsqrt(2.0d0)/2.0d0*kp*(1-y(2))
     #     /(f*kd*y(1)/kt)**0.5d0*f*kd/kt
           dfdy(2,2) = -
dsqrt(2.0d0)*kp*(f*kd*y(1)/kt)**0.5d0
      endif


      return
      end
```

```
c     ******************
c     ******************
c     ******************

      subroutine trapzd(func,a,b,s,n)
      integer n
      double precision a,b,s,func
      external func

c     This routine computes the nth stage of refinement of
      an extended trapezoidal rule. c    func is   input   as
      the name of the function to be integrated between
      limits a and b, c   also input. when called with n=1,
      the routine returns as s the crudest estimate of b

c     a f(x)dx. subsequent
c     calls with n=2,3,... (in that sequential order) will
      improve the accuracy of s by
c     adding 2n-2
c     additional interior points. s should not be modified
      between sequential calls.
      integer it,j
      double precision del,sum,tnm,x
      if (n.eq.1) then
      s=0.5d0*(b-a)*(func(a)+func(b))
      else
      it=2**(n-2)
      tnm=it
      del=(b-a)/tnm       !this is the spacing of the points
                          !to be added.
      x=a+0.5d0*del
      sum=0.0d0
      do j=1,it
      sum=sum+func(x)
      x=x+del
      enddo
      s=0.5d0*(s+(b-a)*sum/tnm) !this replaces s by its
                                !refined value.
      endif
      return
      end

c     ****************


      subroutine qsimp(func,a,b,s)
      integer jmax
```

```
      double precision a,b,func,s,eps
      external func
      parameter (eps=1.d-10, jmax=20)
c     uses trapzd
c     Returns as s the integral of the function func from a
      to b. The parameters eps can be set to the desired
      fractional accuracy and jmax so that 2 to the power
      jmax-1 is the maximum allowed number of steps.
      Integration is performed by Simpson's rule.

      integer j
      double precision os,ost,st
      ost=-1.0d30
      os= -1.0d30
      do j=1,jmax
      call trapzd(func,a,b,st,j)
      s=(4.0d0*st-ost)/3.0d0
      if (j.gt.5) then
      if (abs(s-os).lt.eps*abs(os).or.
     #(s.eq.0..and.os.eq.0.)) return
      endif
      os=s
      ost=st
      enddo
      pause 'too many steps in qsimp'
      end

c     *********

      subroutine
     odeint(ystart,nvar,x1,x2,eps,h1,hmin,nok,nbad,derivs,
     #stifbs)
      integer nbad,nok,nvar,kmaxx,maxstp,nmax
      double precision eps,h1,hmin,x1,x2,ystart(nvar),tiny
      external derivs,stifbs
      parameter (maxstp=100000,nmax=50,kmaxx=200,tiny=1.d-
     #30)

c     Runge-kutta driver with adaptive stepsize control.
      integrate the starting values ystart(1:nvar)     from
      x1 to x2 with accuracy eps, storing intermediate
      results in the common block /path/. h1 should be set
      as a guessed first stepsize, hmin as the minimum
      allowed stepsize (can be zero). On output nok and nbad
      are the number of good and bad (but retried and fixed)
      steps taken, and ystart is replaced by values at the
      end of the integration interval. derivs is the user-
```

```
      supplied subroutine for calculating the right-hand
      side derivative, while rkqs is the name of the stepper
      routine to be used. /path/ contains its own
      information about how often an intermediate value is
      to be stored.

      integer i,kmax,kount,nstp
      double precision
dxsav,h,hdid,hnext,x,xsav,dydx(nmax),xp(kmaxx),
     #y(nmax),yp(nmax,kmaxx),yscal(nmax)
      common /path/ kmax,kount,dxsav,xp,yp

c     User storage for intermediate results. preset dxsav
and kmax.

      x=x1
      h=sign(h1,x2-x1)
      nok=0
      nbad=0
      kount=0
      do  i=1,nvar
      y(i)=ystart(i)
      enddo
      if (kmax.gt.0) xsav=x-2.*dxsav
      do  nstp=1,maxstp !take at most maxstp steps.
      call derivs(x,y,dydx)
      do  i=1,nvar

c     Scaling used to monitor accuracy. this general-purpose
      choice can be modified if need be.

      yscal(i)=abs(y(i))+abs(h*dydx(i))+tiny
      enddo
      if(kmax.gt.0)then
      if(abs(x-xsav).gt.abs(dxsav)) then
      if(kount.lt.kmax-1)then
      kount=kount+1
      xp(kount)=x
      do  i=1,nvar
      yp(i,kount)=y(i)
      enddo
      xsav=x
      endif
      endif
      endif
      if((x+h-x2)*(x+h-x1).gt.0.) h=x2-x
      call stifbs(y,dydx,nvar,x,h,eps,yscal,hdid,
```

```
      #hnext,derivs)
      if(hdid.eq.h)then
      nok=nok+1
      else
      nbad=nbad+1
      endif
      if((x-x2)*(x2-x1).ge.0.)then
      do  i=1,nvar
      ystart(i)=y(i)
      enddo
      if(kmax.ne.0)then
      kount=kount+1 !save final step.
      xp(kount)=x
      do  i=1,nvar
      yp(i,kount)=y(i)
      enddo
      endif
      return
      endif
      if(abs(hnext).lt.hmin) pause
      #'stepsize smaller than minimum in odeint'
      h=hnext
      enddo
      pause 'too many steps in odeint'
      return
      end

c     *******

      subroutine
      stifbs(y,dydx,nv,x,htry,eps,yscal,hdid,hnext,derivs)
      integer nv,nmax,kmaxx,imax
      double precision eps,hdid,hnext,htry
      double precision x,dydx(nv),y(nv),yscal(nv),
      #safe1,safe2,redmax,redmin,tiny,scalmx
      external derivs
      parameter(nmax=50,kmaxx=7,imax=kmaxx+1,safe1=.25d0,saf
      #e2=.7d0,redmax=1.d-5,redmin=.7d0,tiny=1.d-
      #30,scalmx=.1d0)

c     uses derivs,jacobn,simpr,pzextr

c     Semi-implicit extrapolation step for integrating stiff
      o.d.e.’s, with monitoring of local truncation error to
      adjust stepsize. input are the dependent variable
      vector y(1:nv) and its derivative dydx(1:nv) at the
      starting value of the independent variable x. also
```

input are the stepsize to be attempted htry, the required accuracy eps, and the vector yscal(1:nv) against which the error is scaled. on output, y and x are replaced by their newv alues, hdid is the stepsize that was actually accomplished, and hnext is the estimated next stepsize. derivs is a user-supplied subroutine that computes the derivatives of the right-hand side with respect to x, while jacobn (a fixed name) is a user-supplied subroutine that computes the jacobi matrix of derivatives of the right-hand side with respect to the components of y. be sure to set htry on successive steps to the value of hnext returned from the previous step, as is the case if the routine is called by odeint.

```
integer i,iq,k,kk,km,kmax,kopt,nvold,nseq(imax)
double precision eps1,epsold,errmax,fact
double precision h,red,scale,work,wrkmin,
double precision xest,xnew,a(imax),alf(kmaxx,kmaxx)
double precision dfdx(nmax),dfdy(nmax,nmax),
double precison err(kmaxx),yerr(nmax)
double precison ysav(nmax),yseq(nmax)
logical first,reduct
save a,alf,epsold,first,kmax,kopt,nseq,nvold,xnew
data first/.true./,epsold/-1./,nvold/-1/
data nseq /2,6,10,14,22,34,50,70/
if(eps.ne.epsold.or.nv.ne.nvold)then
hnext=-1.d29
xnew=-1.d29
eps1=safe1*eps
a(1)=nseq(1)+1.0d0
do  k=1,kmaxx
a(k+1)=a(k)+nseq(k+1)
enddo
do  iq=2,kmaxx
do  k=1,iq-1.0d0
alf(k,iq)=eps1**((a(k+1)-a(iq+1))/
#((a(iq+1)-a(1)+1.0d0)*(2*k+1)))
enddo
enddo
epsold=eps
nvold=nv
a(1)=nv+a(1)
do  k=1,kmaxx
a(k+1)=a(k)+nseq(k+1)
enddo
do  kopt=2,kmaxx-1
```

```
      if(a(kopt+1).gt.a(kopt)*alf(kopt-1,kopt))goto 1
      enddo
1     kmax=kopt
      endif
      h=htry
      do i=1,nv
      ysav(i)=y(i)
      enddo
      call jacobn(x,y,dfdx,dfdy,nv,nmax)
      if(h.ne.hnext.or.x.ne.xnew)then
      first=.true.
      kopt=kmax
      endif
      reduct=.false.
2     do k=1,kmax
      xnew=x+h
      if(xnew.eq.x)pause 'stepsize underflow in stifbs'
      call simpr(ysav,dydx,dfdx,dfdy,nmax,
     #nv,x,h,nseq(k),yseq,derivs)
      xest=(h/nseq(k))**2
      call pzextr(k,xest,yseq,y,yerr,nv)
      if(k.ne.1)then
      errmax=tiny
      do  i=1,nv
      errmax=max(errmax,abs(yerr(i)/yscal(i)))
      enddo
      errmax=errmax/eps
      km=k-1
      err(km)=(errmax/safe1)**(1.0d0/(2*km+1))
      endif
      if(k.ne.1.and.(k.ge.kopt-1.or.first))then
      if(errmax.lt.1.)goto 4
      if(k.eq.kmax.or.k.eq.kopt+1)then
      red=safe2/err(km)
      goto 3
      else if(k.eq.kopt)then
      if(alf(kopt-1,kopt).lt.err(km))then
      red=1./err(km)
      goto 3
      endif
      else if(kopt.eq.kmax)then
      if(alf(km,kmax-1).lt.err(km))then
      red=alf(km,kmax-1)*
     #safe2/err(km)
      goto 3
      endif
      else if(alf(km,kopt).lt.err(km))then
```

```
      red=alf(km,kopt-1)/err(km)
      goto 3
      endif
      endif
      enddo
3     red=min(red,redmin)
      red=max(red,redmax)
      h=h*red
      reduct=.true.
      goto 2
4     x=xnew
      hdid=h
      first=.false.
      wrkmin=1.d35
      do  kk=1,km
      fact=max(err(kk),scalmx)
      work=fact*a(kk+1)
      if(work.lt.wrkmin)then
      scale=fact
      wrkmin=work
      kopt=kk+1
      endif
      enddo
      hnext=h/scale
      if(kopt.ge.k.and.kopt.ne.kmax.and..not.reduct)then
      fact=max(scale/alf(kopt-1,kopt),scalmx)
      if(a(kopt+1)*fact.le.wrkmin)then
      hnext=h/fact
      kopt=kopt+1
      endif
      endif
      return
      end

c     ******************

      subroutine
      simpr(y,dydx,dfdx,dfdy,nmax,n,xs,htot,nstep,yout,
      #derivs)
      integer n,nmax,nstep,nmaxx
      double precision htot,xs,dfdx(n)
      double precision dfdy(nmax,nmax),dydx(n),y(n),
      #yout(n)
      external derivs
      parameter (nmaxx=50)
c     uses derivs,lubksb,ludcmp
```

```
c     performs one step of semi-implicit midpoint rule.
      input are the dependent variable y(1:n), its
      derivative dydx(1:n), the derivative of the right-hand
      side with respect to x, dfdx(1:n), and the jacobian
      dfdy(1:nmax,1:nmax) at xs. Also input are htot, the
      total step to be taken, and nstep, the number of
      substeps to be used. the output is returned as
      yout(1:n). derivs is the user-supplied subroutine that
      calculates dydx.

      integer i,j,nn,indx(nmaxx)
      double precision d,h,x,a(nmaxx,nmaxx)
      double precision del(nmaxx),ytemp(nmaxx)
      h=htot/nstep
      do i=1,n
      do j=1,n
      a(i,j)=-h*dfdy(i,j)
      enddo
      a(i,i)=a(i,i)+1.
      enddo
      call ludcmp(a,n,nmaxx,indx,d)
      do  i=1,n
      yout(i)=h*(dydx(i)+h*dfdx(i))
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
      do  i=1,n
      del(i)=yout(i)
      ytemp(i)=y(i)+del(i)
      enddo
      x=xs+h
      call derivs(x,ytemp,yout)
      do  nn=2,nstep
      do  i=1,n
      yout(i)=h*yout(i)-del(i)
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
      do  i=1,n
      del(i)=del(i)+2.*yout(i)
      ytemp(i)=ytemp(i)+del(i)
      enddo
      x=x+h
      call derivs(x,ytemp,yout)
      enddo
      do  i=1,n
      yout(i)=h*yout(i)-del(i)
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
```

```
      do  i=1,n
      yout(i)=ytemp(i)+yout(i)
      enddo
      return
      end


c     ***********

      subroutine pzextr(iest,xest,yest,yz,dy,nv)
      integer iest,nv,imax,nmax
      double precision xest,dy(nv),yest(nv),yz(nv)
      parameter (imax=13,nmax=50)

c     Use polynomial extrapolation to evaluate nv functions
      at x = 0 by fitting a polynomial to a sequence of
      estimates with progressively smaller values x = xest,
      and corresponding function vectors yest(1:nv). This
      call is number iest in the sequence of calls.
      Extrapolated function values are output as yz(1:nv),
      and their estimated error is output as dy(1:nv).
      Parameters: maximum expected value of iest is imax; of
      nv is nmax.

      integer j,k1
      double precision delta,f1,f2,q,d(nmax)
      double precision qcol(nmax,imax),x(imax)
      save qcol,x
      x(iest)=xest
      do  j=1,nv
      dy(j)=yest(j)
      yz(j)=yest(j)
      enddo
      if(iest.eq.1) then
      do  j=1,nv
      qcol(j,1)=yest(j)
      enddo
      else
      do  j=1,nv
      d(j)=yest(j)
      enddo
      do  k1=1,iest-1
      delta=1.0/(x(iest-k1)-xest)
      f1=xest*delta
      f2=x(iest-k1)*delta
      do  j=1,nv
      q=qcol(j,k1)
```

```
      qcol(j,k1)=dy(j)
      delta=d(j)-q
      dy(j)=f1*delta
      d(j)=f2*delta
      yz(j)=yz(j)+dy(j)
      enddo
      enddo
      do  j=1,nv
      qcol(j,iest)=dy(j)
      enddo
      endif
      return
      end

c     ******************

      subroutine ludcmp(a,n,np,indx,d)
      integer n,np,indx(n),nmax
      double precision d,a(np,np),tiny
      parameter (nmax=500,tiny=1.0d-20)

c     Given a matrix a(1:n,1:n), with physical dimension np
      by  np,  this  routine  replaces  it  by  the  lu
      decomposition of a rowwise permutation of itself. a
      and n are input. a is output, arranged as in equation
      (2.3.14) above; indx(1:n) is an output vector that
      records the row permutation effected by the partial
      pivoting; d is output as ±1 depending on whether the
      number  of  row  interchanges  was  even  or  odd,
      respectively. This routine is used in combination with
      lubksb to solve linear equations or invert a matrix.

      integer i,imax,j,k
      double precision aamax,dum,sum,vv(nmax)

      d=1.
      do  i=1,n
      aamax=0.0d0
      do  j=1,n
      if (abs(a(i,j)).gt.aamax) aamax=abs(a(i,j))
      enddo
      if (aamax.eq.0.) pause 'singular matrix in ludcmp'
      vv(i)=1./aamax
      enddo
      do  j=1,n
      do  i=1,j-1
      sum=a(i,j)
```

```
      do  k=1,i-1
      sum=sum-a(i,k)*a(k,j)
      enddo
      a(i,j)=sum
      enddo
      aamax=0.0
      do  i=j,n
      sum=a(i,j)
      do  k=1,j-1
      sum=sum-a(i,k)*a(k,j)
      enddo
      a(i,j)=sum
      dum=vv(i)*abs(sum)
      if (dum.ge.aamax) then
      imax=i
      aamax=dum
      endif
      enddo
      if (j.ne.imax)then
      do  k=1,n
      dum=a(imax,k)
      a(imax,k)=a(j,k)
      a(j,k)=dum
      enddo
      d=-d
      vv(imax)=vv(j)
      endif
      indx(j)=imax
      if(a(j,j).eq.0.0d0)a(j,j)=tiny
      if(j.ne.n)then
      dum=1./a(j,j)
      do  i=j+1,n
      a(i,j)=a(i,j)*dum
      enddo
      endif
      enddo
      return
      end

c     ******************

      subroutine lubksb(a,n,np,indx,b)
      integer n,np,indx(n)
      double precision a(np,np),b(n)

c     Solves the set of n linear equations a · x = b. here a
      is input, not as the matrix a but rather as its lu
```

decomposition, determined by the routine ludcmp. Indx is input as the permutation vector returned by ludcmp. b(1:n) is input as the right-hand side vector b, and returns with the solution vector x. a, n, np, and indx are not modified by this routine and can be left in place for successive calls with different right-hand sides b. This routine takes into account the possibility that b will begin with many zero elements, so it is efficient for use in matrix inversion.

```
integer i,ii,j,ll
double precision sum
ii=0
do  i=1,n
ll=indx(i)
sum=b(ll)
b(ll)=b(i)
if (ii.ne.0)then
do  j=ii,i-1
sum=sum-a(i,j)*b(j)
enddo
else if (sum.ne.0.) then
ii=i
endif
b(i)=sum
enddo
do  i=n,1,-1
sum=b(i)
do  j=i+1,n
sum=sum-a(i,j)*b(j)
enddo
b(i)=sum/a(i,i)
enddo
return
end



Program N2_plugs
implicit none

integer                 i,j,nvar,np,nd,nn,n,sum
integer                 nok,nbad,func
integer                 prnt
double precision        lr,lp,ld,lg,vel
double precision        kd,kt,kp,vp,vd,qd,f
double precision        i0,ip,ig,iini,vpi,ii
```

```
      double precision         x0,xp1,xg,xini,xx
      double precision         dydx(2), yout(2), y(2)
      double precision,allocatable::
      didt(:),i1(:),iout(:)
      double precision,allocatable::     iplug(:),idrops(:)
      double precision,allocatable::     ipdum(:),iddum(:)
      double precision,allocatable::
      dxdt(:),x1(:),xout(:)
      double precision,allocatable::     xplug(:),xdrops(:)
      double precision,allocatable::     xpdum(:),xddum(:)
      double precision         tau,ti,tf,tp,td,tg, ttotal
      double precision         pi,area,dia,root,vfactor

c     common blocks

      common /b1/qd,vp,kd
      common /b2/iini,xini
      common /b3/kt,kp,f
      common /b4/vpi
      common /b5/i0,x0
      common /b6/func

c     function declarations

      external p1der, pder,xg,stifbs

c     defines constants.
      prnt=1
      nn=25
      vfactor=1.0d0
      lr=6000.0d0
      lp=1.0d0
      lg=2.0d0
      vel=0.4d0
      kd=2.20d-5
      kp=250.0d0
      kt=6.0d7
      pi=3.14159265359d0
      dia=0.1651d0
      nvar=2
      ti=0.0d0
      f=0.65d0

c     calculates the number of plugs and droplets

      np=lr/(lp+lg)
      nd=np/nn
```

```
c    allocates array sizes

     allocate(didt(np+1),i1(np+1),iout(np+1))
     allocate(iplug(np+1),idrops(nd))
     allocate(ipdum(np+1),iddum(nd))
     allocate(dxdt(np+1),x1(np+1),xout(np+1))
     allocate(xplug(np+1),xdrops(nd))
     allocate(xpdum(np+1),xddum(nd))

c    calculates various parameters


     area=pi*(dia/2.0d0)**2
     vp=lp*area
     vd=(pi*(0.1651d0/2.0d0)**2)/100.0d0*vfactor
     ld=2.0d0*vd
     ld=ld/(4.0d0/3.0d0*pi)
     root=1.0d0/3.0d0
     ld=ld**(root)
     ld=2.0d0*ld
     tau=lr/vel
     tp=lp/vel
     td=ld/vel
     tg=lg/vel
     qd=vd/td


c    sets initial concentrations

     x0=0.0d0
     i0=1.0d0

     ii=i0*dexp(-kd*tau)
     xx=xg(i0,x0,tau)



     i1=0.0d0
     i1(1)=i0
     x1=0.0d0
     idrops=0.0d0
     iddum=idrops
     iplug=0.0d0
     ipdum=iplug
     iplug(1)=i0
     ipdum(1)=i0
```

```
      iout=0.0d0
      xplug=0.0d0
      xdrops=0.0d0
      xddum=xdrops
      xpdum=xplug

c      opens output file.

      open(unit=1,file='n2plugs.out',status='unknown')
      open(unit=2,file='check.out',status='unknown')

c     calculates the concentration of the first plug through
      the   reactor.   also   establishes   location   and
      concentration of droplets.

      plug=1.0
      if(nd.ne.0)then
           sum=0.0d0
           vpi=vp
           do i=1,np
                     if(mod(i,nn).eq.0) then
                          sum=sum+1
                          i1(i+1)=ig(i1(i),tg+tp-td)
                          x1(i+1)=xg(i1(i),x1(i),tg+tp-td)
                          y(1)=i1(i+1)
                          y(2)=x1(i+1)
                          call odeint(y,nvar,ti,td,1.0d-
#                         8,1.0d-8,0.0d0,
#                         nok,nbad,p1der,stifbs)
                          x1(i+1)=y(2)
                          i1(i+1)=y(1)
                          idrops(sum)=i1(i+1)
                          if(vpi.gt.vp-(vp/vd-1)*vd)then
                          vpi=vp-vd*sum
                          endif
                     else
                          i1(i+1)=ig(i1(i),tg+tp)
                          x1(i+1)=xg(i1(i),x1(i),tg+tp)
                     endif
           enddo
      endif
      iplug=i1
      xplug=x1
      ipdum=iplug
      xpdum=xplug
      iddum=idrops
      xddum=xdrops
```

```
      vpi=vp

      write(1,*)ii,xx
      write(1,*)1.0d0,i1(np+1),x1(np+1)

c     determines the concentration and conversion for each
      of the plugs in the reactor, pi, i > 1.

      do i=1,np/10
           sum=0
           if(nd.eq.0)then
                do j=1,np
                ipdum(j+1)=ig(ipdum(j),tg+tp)
                write(1,*)ipdum(j+1)
                enddo
           else
                do j=1,np
                vp=vpi
                if(mod(j,nn).eq.0) then
                sum=sum+1
                y(1)=ig(ipdum(j),tg-td)
                y(2)=xg(ipdum(j),xpdum(j),tg-td)
                iddum(sum)=ig(idrops(sum),tg-td)
                iini=idrops(sum)
                call odeint(y,nvar,ti,tp+td,1.0d
#               8,1.0d-8,0.0d0,
#               nok,nbad,pder,stifbs)
                ipdum(j+1)=y(1)
                xpdum(j+1)=y(2)
                iddum(sum)=ipdum(j+1)
                else
                ipdum(j+1)=ig(ipdum(j),tg+tp)
                xpdum(j+1)=xg(ipdum(j),xpdum(j),tg+tp)
                endif

                enddo
           endif
           iplug=ipdum
           idrops=iddum
           xplug=xpdum
           xdrops=xddum
           if(mod(i,prnt).eq.0) then

      write(*,*)((tp+tg)*i+tau)/tau,iplug(np+1),xplug(np+1)
      write(1,*)((tp+tg)*i+tau)/tau,iplug(np+1),xplug(np+1)

           endif
```

```
      enddo

      end

c     ****subroutines and functions.*****

      double precision function ig(i0,tg)
      implicit none

      double precision i0,tg
      double precision qd,vp,kd

      common /b1/qd,vp,kd

      ig=i0*dexp(-kd*tg)

      return
      end

c     ********
      double precision function ip(i0,vpi,t)
      implicit none

      double precision i0,vpi,t
      double precision qd,vp,kd

      common /b1/qd,vp,kd

      ip=i0*dexp(-kd*t)*(vpi-qd*t)

      return
      end

c     ********
      double precision function xg(i0,x0,t)
      implicit none

      double precision t,func
      double precision i0,x0,xout
      double precision qd,vp,kd
      double precision iini,xini
      double precision kt,kp,f
      external func

      common /b1/qd,vp,kd
      common /b2/iini,xini
      common /b3/kt,kp,f
```

```
      iini=i0
      xini=x0

      call qsimp(func,0.0d0,t,xout)
      xg=1.0d0-(1.0d0-xini)*dexp(-xout)


      return
      end

c     **********

      subroutine pder(t,y,dydx)
      implicit none

      double precision    t,iini,xini
      double precision    dydx(2),y(2)
      double precision    qd,vp,kd
      double precision    kt,kp,f

      common /b1/qd,vp,kd
      common /b2/iini,xini
      common /b3/kt,kp,f

      dydx(1)=-(qd/vp+kd)*y(1)+qd/vp*iini
      dydx(2)=kp*(1.0d0-y(2))*((2.0d0*f*kd*y(1))/kt)**0.5d0

      return
      end

c     *********

      subroutine p1der(t,y,dydx)
      implicit none

      double precision    t
      double precision    dydx(2),y(2)
      double precision    qd,vp,kd
      double precision    kt,kp,f
      double precision    vpi

      common /b1/qd,vp,kd
      common /b3/kt,kp,f
      common /b4/vpi

      dydx(1)=-(qd/vpi+kd)*y(1)
```

```
      dydx(2)=kp*(1.0d0-y(2))*((2.0d0*f*kd*y(1))/kt)**0.5d0

      return
      end

c     ******************

      subroutine jacobn(x,y,dfdx,dfdy,n,nmax)
      implicit none

c     returns the jacobian of the system of differential
equations.
c     used by stifbs

      integer                        i,j,n,nmax,func
      double precision
      x,y(n),dfdx(n),dfdy(nmax,nmax)
      double precision          qd,vp,kd,vpi
      double precision          kt,kp,f


      common /b1/qd,vp,kd
      common /b3/kt,kp,f
      common /b4/vpi
      common /b6/func

      if(func.eq.1)then !jacobian for the gap
           dfdy(1,1) = -kd
           dfdy(1,2) = 0.0d0
           dfdy(2,1) = dsqrt(2.0d0)/2.0d0*kp*(1-y(2))
     #     /(f*kd*y(1)/kt)**0.5d0*f*kd/kt
           dfdy(2,2) = -
dsqrt(2.0d0)*kp*(f*kd*y(1)/kt)**0.5d0
      else if(func.eq.2)then !jacobian for droplet exit
           dfdy(1,1) = -kd-qd/vpi
           dfdy(1,2) = 0.0d0
           dfdy(2,1) = dsqrt(2.0d0)/2.0d0*kp*(1-y(2))
     #     /(f*kd*y(1)/kt)**0.5d0*f*kd/kt
           dfdy(2,2) = -
dsqrt(2.0d0)*kp*(f*kd*y(1)/kt)**0.5d0
      endif


      return
      end

c     ******************
```

```
c     ******************
c     ******************

      subroutine trapzd(func,a,b,s,n)
      integer n
      double precision a,b,s,func
      external func

c     This routine computes the nth stage of refinement of
      an extended trapezoidal rule. c    func is   input   as
      the name of the function to be integrated between
      limits a and b, c   also input. when called with n=1,
      the routine returns as s the crudest estimate of b

c     a f(x)dx. subsequent
c     calls with n=2,3,... (in that sequential order) will
      improve the accuracy of s by
c     adding 2n-2
c     additional interior points. s should not be modified
      between sequential calls.
      integer it,j
      double precision del,sum,tnm,x
      if (n.eq.1) then
      s=0.5d0*(b-a)*(func(a)+func(b))
      else
      it=2**(n-2)
      tnm=it
      del=(b-a)/tnm       !this is the spacing of the points
                          !to be added.
      x=a+0.5d0*del
      sum=0.0d0
      do j=1,it
      sum=sum+func(x)
      x=x+del
      enddo
      s=0.5d0*(s+(b-a)*sum/tnm) !this replaces s by its
                                !refined value.
      endif
      return
      end

c     ****************


      subroutine qsimp(func,a,b,s)
      integer jmax
      double precision a,b,func,s,eps
```

```
      external func
      parameter (eps=1.d-10, jmax=20)

c     uses trapzd
c     Returns as s the integral of the function func from a
      to b. The parameters eps can be set to the desired
      fractional accuracy and jmax so that 2 to the power
      jmax-1 is the maximum allowed number of steps.
      Integration is performed by Simpson's rule.

      integer j
      double precision os,ost,st
      ost=-1.0d30
      os= -1.0d30
      do j=1,jmax
      call trapzd(func,a,b,st,j)
      s=(4.0d0*st-ost)/3.0d0
      if (j.gt.5) then
      if (abs(s-os).lt.eps*abs(os).or.
     #(s.eq.0..and.os.eq.0.)) return
      endif
      os=s
      ost=st
      enddo
      pause 'too many steps in qsimp'
      end

c     *********

      subroutine
     odeint(ystart,nvar,x1,x2,eps,h1,hmin,nok,nbad,derivs,
     #stifbs)
      integer nbad,nok,nvar,kmaxx,maxstp,nmax
      double precision eps,h1,hmin,x1,x2,ystart(nvar),tiny
      external derivs,stifbs
      parameter (maxstp=100000,nmax=50,kmaxx=200,tiny=1.d-
     #30)

c     Runge-kutta driver with adaptive stepsize control.
      integrate the starting values ystart(1:nvar)     from
      x1 to x2 with accuracy eps, storing intermediate
      results in the common block /path/. h1 should be set
      as a guessed first stepsize, hmin as the minimum
      allowed stepsize (can be zero). On output nok and nbad
      are the number of good and bad (but retried and fixed)
      steps taken, and ystart is replaced by values at the
      end of the integration interval. derivs is the user-
```

```
      supplied subroutine for calculating the right-hand
      side derivative, while rkqs is the name of the stepper
      routine to be used. /path/ contains its own
      information about how often an intermediate value is
      to be stored.

      integer i,kmax,kount,nstp
      double precision
dxsav,h,hdid,hnext,x,xsav,dydx(nmax),xp(kmaxx),
     #y(nmax),yp(nmax,kmaxx),yscal(nmax)
      common /path/ kmax,kount,dxsav,xp,yp

c     User storage for intermediate results. preset dxsav
and kmax.

      x=x1
      h=sign(h1,x2-x1)
      nok=0
      nbad=0
      kount=0
      do  i=1,nvar
      y(i)=ystart(i)
      enddo
      if (kmax.gt.0) xsav=x-2.*dxsav
      do  nstp=1,maxstp !take at most maxstp steps.
      call derivs(x,y,dydx)
      do  i=1,nvar

c     Scaling used to monitor accuracy. this general-purpose
      choice can be modified if need be.

      yscal(i)=abs(y(i))+abs(h*dydx(i))+tiny
      enddo
      if(kmax.gt.0)then
      if(abs(x-xsav).gt.abs(dxsav)) then
      if(kount.lt.kmax-1)then
      kount=kount+1
      xp(kount)=x
      do  i=1,nvar
      yp(i,kount)=y(i)
      enddo
      xsav=x
      endif
      endif
      endif
      if((x+h-x2)*(x+h-x1).gt.0.) h=x2-x
      call stifbs(y,dydx,nvar,x,h,eps,yscal,hdid,
```

```
      #hnext,derivs)
      if(hdid.eq.h)then
      nok=nok+1
      else
      nbad=nbad+1
      endif
      if((x-x2)*(x2-x1).ge.0.)then
      do  i=1,nvar
      ystart(i)=y(i)
      enddo
      if(kmax.ne.0)then
      kount=kount+1 !save final step.
      xp(kount)=x
      do  i=1,nvar
      yp(i,kount)=y(i)
      enddo
      endif
      return
      endif
      if(abs(hnext).lt.hmin) pause
      #'stepsize smaller than minimum in odeint'
      h=hnext
      enddo
      pause 'too many steps in odeint'
      return
      end

c     *******

      subroutine
      stifbs(y,dydx,nv,x,htry,eps,yscal,hdid,hnext,derivs)
      integer nv,nmax,kmaxx,imax
      double precision eps,hdid,hnext,htry
      double precision x,dydx(nv),y(nv),yscal(nv),
      #safe1,safe2,redmax,redmin,tiny,scalmx
      external derivs
      parameter(nmax=50,kmaxx=7,imax=kmaxx+1,safe1=.25d0,saf
      #e2=.7d0,redmax=1.d-5,redmin=.7d0,tiny=1.d-
      #30,scalmx=.1d0)

c     uses derivs,jacobn,simpr,pzextr

c     Semi-implicit extrapolation step for integrating stiff
      o.d.e.’s, with monitoring of local truncation error to
      adjust stepsize. input are the dependent variable
      vector y(1:nv) and its derivative dydx(1:nv) at the
      starting value of the independent variable x. also
```

input are the stepsize to be attempted htry, the required accuracy eps, and the vector yscal(1:nv) against which the error is scaled. on output, y and x are replaced by their newv alues, hdid is the stepsize that was actually accomplished, and hnext is the estimated next stepsize. derivs is a user-supplied subroutine that computes the derivatives of the right-hand side with respect to x, while jacobn (a fixed name) is a user-supplied subroutine that computes the jacobi matrix of derivatives of the right-hand side with respect to the components of y. be sure to set htry on successive steps to the value of hnext returned from the previous step, as is the case if the routine is called by odeint.

```
integer i,iq,k,kk,km,kmax,kopt,nvold,nseq(imax)
double precision eps1,epsold,errmax,fact
double precision h,red,scale,work,wrkmin,
double precision xest,xnew,a(imax),alf(kmaxx,kmaxx)
double precision dfdx(nmax),dfdy(nmax,nmax),
double precison err(kmaxx),yerr(nmax)
double precison ysav(nmax),yseq(nmax)
logical first,reduct
save a,alf,epsold,first,kmax,kopt,nseq,nvold,xnew
data first/.true./,epsold/-1./,nvold/-1/
data nseq /2,6,10,14,22,34,50,70/
if(eps.ne.epsold.or.nv.ne.nvold)then
hnext=-1.d29
xnew=-1.d29
eps1=safe1*eps
a(1)=nseq(1)+1.0d0
do  k=1,kmaxx
a(k+1)=a(k)+nseq(k+1)
enddo
do  iq=2,kmaxx
do  k=1,iq-1.0d0
alf(k,iq)=eps1**((a(k+1)-a(iq+1))/
#((a(iq+1)-a(1)+1.0d0)*(2*k+1)))
enddo
enddo
epsold=eps
nvold=nv
a(1)=nv+a(1)
do  k=1,kmaxx
a(k+1)=a(k)+nseq(k+1)
enddo
do  kopt=2,kmaxx-1
```

```
      if(a(kopt+1).gt.a(kopt)*alf(kopt-1,kopt))goto 1
      enddo
1     kmax=kopt
      endif
      h=htry
      do i=1,nv
      ysav(i)=y(i)
      enddo
      call jacobn(x,y,dfdx,dfdy,nv,nmax)
      if(h.ne.hnext.or.x.ne.xnew)then
      first=.true.
      kopt=kmax
      endif
      reduct=.false.
2     do k=1,kmax
      xnew=x+h
      if(xnew.eq.x)pause 'stepsize underflow in stifbs'
      call simpr(ysav,dydx,dfdx,dfdy,nmax,
     #nv,x,h,nseq(k),yseq,derivs)
      xest=(h/nseq(k))**2
      call pzextr(k,xest,yseq,y,yerr,nv)
      if(k.ne.1)then
      errmax=tiny
      do  i=1,nv
      errmax=max(errmax,abs(yerr(i)/yscal(i)))
      enddo
      errmax=errmax/eps
      km=k-1
      err(km)=(errmax/safe1)**(1.0d0/(2*km+1))
      endif
      if(k.ne.1.and.(k.ge.kopt-1.or.first))then
      if(errmax.lt.1.)goto 4
      if(k.eq.kmax.or.k.eq.kopt+1)then
      red=safe2/err(km)
      goto 3
      else if(k.eq.kopt)then
      if(alf(kopt-1,kopt).lt.err(km))then
      red=1./err(km)
      goto 3
      endif
      else if(kopt.eq.kmax)then
      if(alf(km,kmax-1).lt.err(km))then
      red=alf(km,kmax-1)*
     #safe2/err(km)
      goto 3
      endif
      else if(alf(km,kopt).lt.err(km))then
```

```
      red=alf(km,kopt-1)/err(km)
      goto 3
      endif
      endif
      enddo
3     red=min(red,redmin)
      red=max(red,redmax)
      h=h*red
      reduct=.true.
      goto 2
4     x=xnew
      hdid=h
      first=.false.
      wrkmin=1.d35
      do  kk=1,km
      fact=max(err(kk),scalmx)
      work=fact*a(kk+1)
      if(work.lt.wrkmin)then
      scale=fact
      wrkmin=work
      kopt=kk+1
      endif
      enddo
      hnext=h/scale
      if(kopt.ge.k.and.kopt.ne.kmax.and..not.reduct)then
      fact=max(scale/alf(kopt-1,kopt),scalmx)
      if(a(kopt+1)*fact.le.wrkmin)then
      hnext=h/fact
      kopt=kopt+1
      endif
      endif
      return
      end

c     ******************

      subroutine
      simpr(y,dydx,dfdx,dfdy,nmax,n,xs,htot,nstep,yout,
      #derivs)
      integer n,nmax,nstep,nmaxx
      double precision htot,xs,dfdx(n)
      double precision dfdy(nmax,nmax),dydx(n),y(n),
      #yout(n)
      external derivs
      parameter (nmaxx=50)
c     uses derivs,lubksb,ludcmp
```

```
c     performs one step of semi-implicit midpoint rule.
      input are the dependent variable y(1:n), its
      derivative dydx(1:n), the derivative of the right-hand
      side with respect to x, dfdx(1:n), and the jacobian
      dfdy(1:nmax,1:nmax) at xs. Also input are htot, the
      total step to be taken, and nstep, the number of
      substeps to be used. the output is returned as
      yout(1:n). derivs is the user-supplied subroutine that
      calculates dydx.

      integer i,j,nn,indx(nmaxx)
      double precision d,h,x,a(nmaxx,nmaxx)
      double precision del(nmaxx),ytemp(nmaxx)
      h=htot/nstep
      do i=1,n
      do j=1,n
      a(i,j)=-h*dfdy(i,j)
      enddo
      a(i,i)=a(i,i)+1.
      enddo
      call ludcmp(a,n,nmaxx,indx,d)
      do  i=1,n
      yout(i)=h*(dydx(i)+h*dfdx(i))
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
      do  i=1,n
      del(i)=yout(i)
      ytemp(i)=y(i)+del(i)
      enddo
      x=xs+h
      call derivs(x,ytemp,yout)
      do  nn=2,nstep
      do  i=1,n
      yout(i)=h*yout(i)-del(i)
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
      do  i=1,n
      del(i)=del(i)+2.*yout(i)
      ytemp(i)=ytemp(i)+del(i)
      enddo
      x=x+h
      call derivs(x,ytemp,yout)
      enddo
      do  i=1,n
      yout(i)=h*yout(i)-del(i)
      enddo
      call lubksb(a,n,nmaxx,indx,yout)
```

```
      do  i=1,n
      yout(i)=ytemp(i)+yout(i)
      enddo
      return
      end


c     ***********

      subroutine pzextr(iest,xest,yest,yz,dy,nv)
      integer iest,nv,imax,nmax
      double precision xest,dy(nv),yest(nv),yz(nv)
      parameter (imax=13,nmax=50)

c     Use polynomial extrapolation to evaluate nv functions
      at x = 0 by fitting a polynomial to a sequence of
      estimates with progressively smaller values x = xest,
      and corresponding function vectors yest(1:nv). This
      call is number iest in the sequence of calls.
      Extrapolated function values are output as yz(1:nv),
      and their estimated error is output as dy(1:nv).
      Parameters: maximum expected value of iest is imax; of
      nv is nmax.

      integer j,k1
      double precision delta,f1,f2,q,d(nmax)
      double precision qcol(nmax,imax),x(imax)
      save qcol,x
      x(iest)=xest
      do  j=1,nv
      dy(j)=yest(j)
      yz(j)=yest(j)
      enddo
      if(iest.eq.1) then
      do  j=1,nv
      qcol(j,1)=yest(j)
      enddo
      else
      do  j=1,nv
      d(j)=yest(j)
      enddo
      do  k1=1,iest-1
      delta=1.0/(x(iest-k1)-xest)
      f1=xest*delta
      f2=x(iest-k1)*delta
      do  j=1,nv
      q=qcol(j,k1)
```

```
      qcol(j,k1)=dy(j)
      delta=d(j)-q
      dy(j)=f1*delta
      d(j)=f2*delta
      yz(j)=yz(j)+dy(j)
      enddo
      enddo
      do  j=1,nv
      qcol(j,iest)=dy(j)
      enddo
      endif
      return
      end

c     ******************

      subroutine ludcmp(a,n,np,indx,d)
      integer n,np,indx(n),nmax
      double precision d,a(np,np),tiny
      parameter (nmax=500,tiny=1.0d-20)

c     Given a matrix a(1:n,1:n), with physical dimension np
      by  np,  this  routine  replaces  it  by  the  lu
      decomposition of a rowwise permutation of itself. a
      and n are input. a is output, arranged as in equation
      (2.3.14) above; indx(1:n) is an output vector that
      records the row permutation effected by the partial
      pivoting; d is output as ±1 depending on whether the
      number  of  row  interchanges  was  even  or  odd,
      respectively. This routine is used in combination with
      lubksb to solve linear equations or invert a matrix.

      integer i,imax,j,k
      double precision aamax,dum,sum,vv(nmax)

      d=1.
      do  i=1,n
      aamax=0.0d0
      do  j=1,n
      if (abs(a(i,j)).gt.aamax) aamax=abs(a(i,j))
      enddo
      if (aamax.eq.0.) pause 'singular matrix in ludcmp'
      vv(i)=1./aamax
      enddo
      do  j=1,n
      do  i=1,j-1
      sum=a(i,j)
```

```
      do  k=1,i-1
      sum=sum-a(i,k)*a(k,j)
      enddo
      a(i,j)=sum
      enddo
      aamax=0.0
      do  i=j,n
      sum=a(i,j)
      do  k=1,j-1
      sum=sum-a(i,k)*a(k,j)
      enddo
      a(i,j)=sum
      dum=vv(i)*abs(sum)
      if (dum.ge.aamax) then
      imax=i
      aamax=dum
      endif
      enddo
      if (j.ne.imax)then
      do  k=1,n
      dum=a(imax,k)
      a(imax,k)=a(j,k)
      a(j,k)=dum
      enddo
      d=-d
      vv(imax)=vv(j)
      endif
      indx(j)=imax
      if(a(j,j).eq.0.0d0)a(j,j)=tiny
      if(j.ne.n)then
      dum=1./a(j,j)
      do  i=j+1,n
      a(i,j)=a(i,j)*dum
      enddo
      endif
      enddo
      return
      end

c     ******************

      subroutine lubksb(a,n,np,indx,b)
      integer n,np,indx(n)
      double precision a(np,np),b(n)

c     Solves the set of n linear equations a · x = b. here a
      is input, not as the matrix a but rather as its lu
```

decomposition, determined by the routine ludcmp. Indx is input as the permutation vector returned by ludcmp. b(1:n) is input as the right-hand side vector b, and returns with the solution vector x. a, n, np, and indx are not modified by this routine and can be left in place for successive calls with different right-hand sides b. This routine takes into account the possibility that b will begin with many zero elements, so it is efficient for use in matrix inversion.

```
integer i,ii,j,ll
double precision sum
ii=0
do  i=1,n
ll=indx(i)
sum=b(ll)
b(ll)=b(i)
if (ii.ne.0)then
do  j=ii,i-1
sum=sum-a(i,j)*b(j)
enddo
else if (sum.ne.0.) then
ii=i
endif
b(i)=sum
enddo
do  i=n,1,-1
sum=b(i)
do  j=i+1,n
sum=sum-a(i,j)*b(j)
enddo
b(i)=sum/a(i,i)
enddo
return
end
```